A Database Publication

AppleUser

Vol. 7 No. 6 June 1987 £1.50

Turn your Apple into a radio teletype receiver

Is a LaserWriter really worth the money?

First impressions of the Macintosh SE

Finding new ways to enhance Appleworks

The WIMPs that link the Mac and the IIgs

Honorware: What to try before you buy

UTILITIES

Enhanced machine code tracer; automated subroutine maker; easier debugging of on-screen listings; updating Dos disc sectors

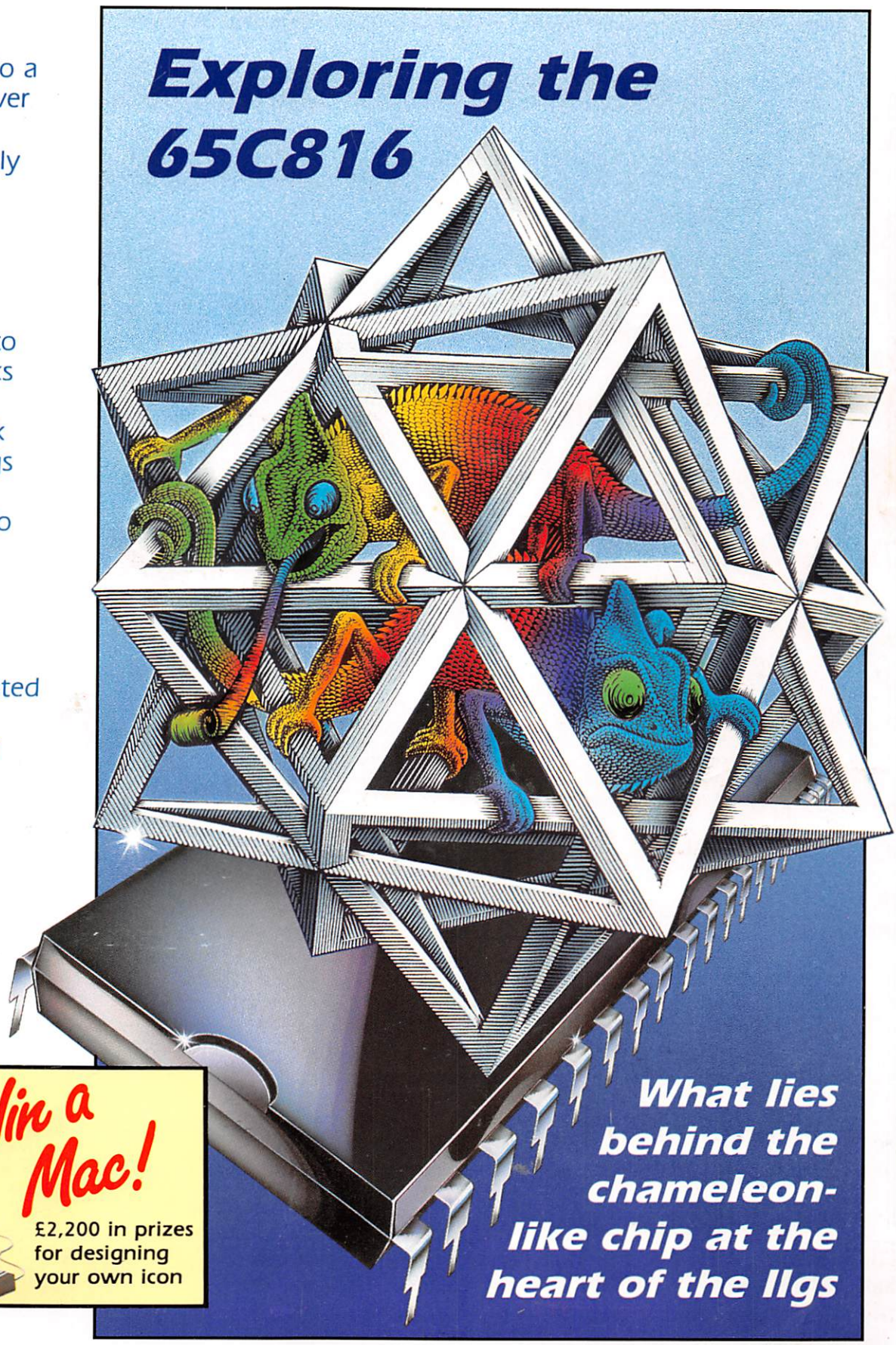
REVIEWS

Microsoft's Works; ObjectLogo + all the latest games

Win a Mac!

£2,200 in prizes for designing your own icon

Exploring the 65C816



What lies behind the chameleon-like chip at the heart of the IIgs

More program power!

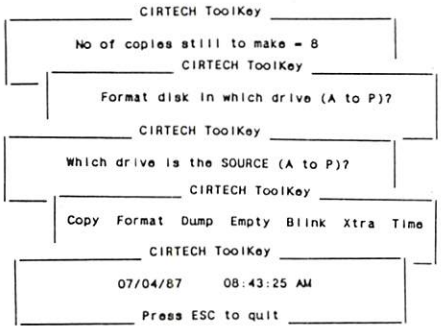
A sophisticated, new operating system for the IIGS

The CIRTECH CP/M Plus System lets you use the huge range of CP/M programs, like Wordstar and dBASE, on your Apple IIGS. The CP/M Plus System comprises a lightning-fast, co-processor card and the most advanced version of the CP/M Plus Operating System software specially designed to fully utilise the powerful features of the GS.

The compact hardware card plugs into one of the standard GS expansion slots and has been designed with a fast 8 MegaHertz Z80H microprocessor to boost the speed of your programs – and *for extra speed*, the GS operates in fast mode with the CP/M Plus System!

The CP/M Plus Operating System is full of versatile, user-friendly features. Special *ToolKey* utilities instantly pop up in a unique window display and you can use them all **at any time**, *even in the middle of running a program!*

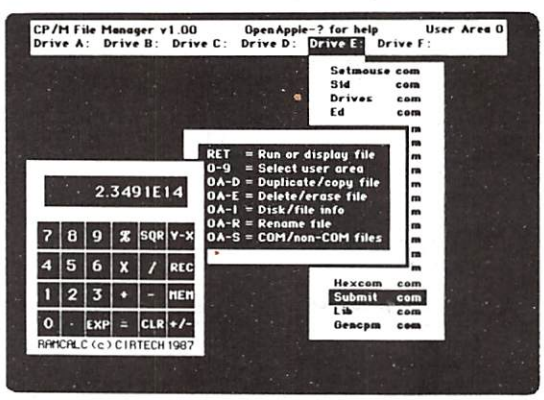
- **COPY** and **FORMAT** are inbuilt disk formatter and copier functions for all standard types of disks (3.5, Profile, SCSI, 5.25, etc.) – *no more problems running out of disk space in the middle of a program!*
- **DUMP** lets you print an instant 'snapshot' of the current text screen at any time.
- **EMPTY** clears the internal printer and auxilliary output buffers. *Yes, the CIRTECH CP/M Plus System has an inbuilt printer buffer (spooler) 12K in size* – that's enough for about 20 A4 pages!



- **BLINK** controls the cursor – *you have the choice of blinking or static!*
- **XTRA** lets you print multiple copies of everything that's in the printer buffer at any time during a CP/M program; *and printing is in 'background mode', so you save time too!*
- **TIME** instantly displays a neat, on-screen window giving you the current time and date – *no excuses for being late now!*

FILE MANAGER

The unique File Manager, with its clear, pull-down windows, lets you see exactly what programs and files are on each disk. *Not only that, you can use the File Manager to run or display files, select user areas, and copy, delete or rename files – using CP/M has never been faster or easier!*



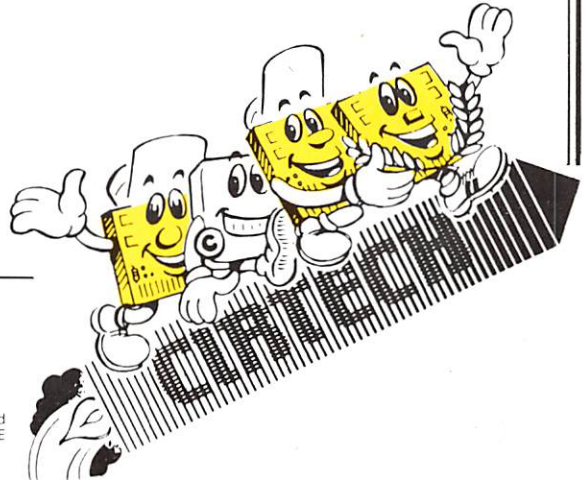
RAMCALC

The CIRTECH CP/M Plus System also features **RAMCALC**, an *ON-SCREEN, FULL FUNCTION CALCULATOR* which you can call up instantly any time you want from within any CP/M program. The calculator has all the normal arithmetic functions *plus* percentage, square root and memory! *And there's no problem if you put RAMCALC away without noting the answer, just call it back and it appears again instantly, exactly as you left it, right down to remembering what's in memory!*

The CP/M Plus System also lets you use the AppleMouse with any CP/M program or change the Mouse control characters with the 'SETMOUSE' utility. The System is fully compatible with all standard CP/M programs and is supplied with over 40 utility programs, including extensive disk-based 'Help' files. All Apple-standard devices such as UniDisk, Disk II 5.25 drives, 3.5 drives and ProFile or SCSI hard disk drives are fully supported – *you can even use ProDOS and CP/M Plus on the same hard disk!* The System is also fully compatible with plusRAM-GS and other Apple standard memory expansion cards.

FOR TOP PERFORMANCE AND SPEED – CIRTECH CP/M PLUS – ONLY £118.00

Also available for the 128K //e and //c (3.5 or 5.25 disk format)



(Prices exclude VAT)

CIRTECH (UK) LIMITED, Currie Road Industrial Estate, Galashiels
Selkirkshire, Scotland, TD1 2BP Telephone (0896) 57790
Telex 265871 (Attn. 84:CPD001)–Telecom Gold Source Mailbox–AAH555

Vol. 7 No. 6 June 1987

Managing Editor
Derek Meakin
Technical Editors
Max Parrott
Cliff McKnight
Production Editor
Peter Glover
Advertising Sales
John Snowden
Gail Blincow
Art Editor
Heather Sheldrick
Reviews Editor
Christopher Payne
News Editor
Mike Cowley

Tel: 061-456 8383 (Editorial)
061-456 8500 (Advertising)
061-480 0171 (Subscriptions)
Telex: 265871 MONREF G
Quoting Ref. 72:MAG001
Telecom Gold: 72:MAG001
Prestel Mailbox: 614568383

Published by:
Database Publications Ltd,
Europa House, 68 Chester Road,
Hazel Grove, Stockport SK7 5NY.

Subscription rates for
12 issues, post free:

£15 UK & Eire (sterling)
£23 Europe
£38 Overseas (airmail)

ABC 11,780
January-June 1986

Writing for Apple User: Articles and programs relating to the Apple are welcome. Articles should preferably be typed or computer-printed, using double spacing. Unsolicited manuscripts, discs etc, should be accompanied by a self addressed stamped envelope, otherwise their return cannot be guaranteed. Unless agreed otherwise, material is accepted on an all rights basis.

News trade distribution: Europress Sales and Distribution Limited, Unit 1, Burgess Road, Ivyhouse Lane, Hastings, East Sussex TN35 4NR. Tel: 0424 430422.

Features

Reviews

Programming

Utilities

Communicate via MicroLink. See Page 27

NEW PRICE DEAL FROM PEANUT

BEST SELLERS

IIe EXTENDED 80 column card lowest ever price	£ 26.00
II+ 80 COLUMN card with built-in soft switch	£ 38.00
Z80 CARD runs all CP/M lowest ever price	£ 29.00
16K RAM card	£ 29.00
PARALLEL printer card including cable	£ 28.00
PRINT-BUFFER card, 32K including cable	£ 74.00
DISK controller card	£ 26.00
PEANUT SLIM disk drive	£ 92.00
SHINWA CPA 80+ printer with NLQ – lowest ever price	£175.00
CIRTECH PLUS RAM XTRA 1Mb	£189.00
CIRTECH CHAMPION printer card lowest ever price	£ 39.00

CIRTECH RANGE

NEW LOW PRICES – DISCOUNT PRICES

Champion printer card	£ 39.00
Champion printer card + 64K buffer	£ 89.00
Cache box, 256K Buffer Parallel	£ 160.00
PLUS RAM XTRA 1Mb	£ 189.00
80 column for IIe with 64K mem	£ 39.00
Z80 card for CP/M	£ 39.00
IIc Z80 module for CP/M	£ 54.00
New 256 Memory Card Expandable to 1MB	£ 89.00

PRINTERS

EPSON LX 86, friction feed	£235.00
LX86, tractor/friction	£254.00
NEW FX85 with NLQ	£ 355.00
BROTHER 1509	£ 399.00
SHINWA CPA 80+, with NLQ	£175.00
CPB80 matrix IBM	£249.00

MONITORS

PHILIPS Monitor 80 12" screen – Green	£ 79.00
Amber	£ 83.00

NOW IN STOCK

We have a limited number of second hand IIe & II+ Systems available. Ring for Prices.

PLUG IN CARDS

80 COLUMN for II+ with built-in softswitch. Auto 40/80 switching. With inverse video chip ..... £ 38.00

80 COLUMN 64K extended for IIe, excellent lowest ever value ..... £ 26.00

16K RAM adds 16K to 48K computers, CP/M and Pascal compatible ..... £ 29.00

Z80 runs all CP/M. This is the card for all that CP/M software ..... £ 29.00

PARALLEL PRINTER, centronics, all control codes and graphic dump feature, includes cable ..... £ 28.00

DISK DRIVE CONTROLLER runs 13 or 16 sector disks automatically ..... £ 26.00

PAL TV COLOUR, composite VHF TV signal and monitor outputs ..... £ 42.00

GRAPPLER compatible printer cards, printer selection switches, with cable.

No buffer ..... £ 47.00

64K buffer ..... £ 87.00

128K RAM blockbuster extra RAM – can increase spreadsheet memory by 128K ..... £ 79.00

IC TESTER plug into remote ZIF socket. Tests over 500 74 and 4000 IC's ..... £ 99.00

MEMORY CHIP TESTER similar to IC Tester, but tests all common memory chips and programs EPROMS and PROMS £119.00

PRINTER/BUFFER 32K, Prints and Buffers. Grappler CP/M, Appleworks etc, compatible. All control codes, graphic dump ..... £ 74.00

EPROM PROGRAMMER programs all EPROMS up to 64K. Compares, verifies, copies, reads ROM-based software included ..... £ 47.00

PIGEON MODEM Full RS232 Modem Ready-to-go multi-format card 300/1200 baud, auto-dial, auto-answer, with software £155.00

DATA HIGHWAY software for the Pigeon Modem. This is a special version of the popular comms package to suit our Modem ..... £ 69.00

ACCELERATOR FOR II+, speeds up the computer 3.5 times. This card will pay for itself very quickly in time saved £175.00

IEEE 488 laboratory interface. Implementation of this popular protocol for the Apple ..... £ 79.00

Time II Clock Card ..... £ 42.00

I/O FOUR PORT uses two versatile 6522 chips to provide four 8 bit ports, timers, serial I/O, counter channels. Excellent value ... £ 32.00

NEW NEW NEW

Clock Card II+, IIe DOS 3.3/PRO DOS/ PASCAL Compatible Outstanding Value £39.00

Intelligent high speed Eprom Programmer Card. Handles up to 27256 Fully automatic operation. Programs for chips at once. £135.00

DISK DRIVES

PDD4 Slim Disk Drive, 140K ..... £ 92.00

PDD6 Apple IIc compatible £105.00

*Also fits Apple GS

(As reviewed in Apple User)

PDD8 Double sided, double density, 640K capacity, inc. patch s/w for Pascal 1.1, CP/M 56, CP/M 60, Prodos 1.1.1, Diversidos, Dos 3.3 ..... £270.00

HARDWARE

Disk Notcher	£ 4.95
Disk storage box (100 disks)	£ 12.95
TILT/SWIVEL monitor base, 12"	£ 12.95
MONITOR Top File Tray	£ 7.95
KEYBOARD 96 key detached unit, 24 function keys IBM style, II+ compatible	£ 89.00
JOYSTICK new all metal, self centreing, fine trim IIe or II+, state which	£ 19.50
ASAD super joystick, fits IBM or Apple, optional self-centreing, fine trim	£ 24.50
POWER SUPPLY 7.5A heavy duty	£ 59.50
Cooling fan clip-on	£ 23.00
PEANUT DISKETTES, Fuji-brand SSDD top quality 5 year guarantee. Packs of 10 (add £1.00 carriage)	£ 13.95
25 Pack unbranded SS/DD	£ 22.50

SOFTWARE

MICROLEDGER the perfect integrated accounts package for the small business, Microledger 1 Dos 3.3

Low price exclusive to Peanut ..... £ 95.00

Microledger 2 enhanced CP/M ..... £ 95.00

Microledger 2, MS-DOS, PC-DOS ..... £ 95.00

FORMAT-80 ENHANCED the ultimate word processor – both Prodos and DOS 3.3, now with spreadsheet ..... **New Low Price** £ 89.00

FLASHCALC the new advanced spreadsheet program from the makers of Visicalc. Compatible with Visicalc files.

For IIe, IIc ..... £ 69.00

Many other programs available – check prices

PEANUT HARD DRIVE

20 Mbyte capacity unit running all DOS 3.3 CP/M, Pascal & PRODOS programs.

New low price £650.00

Tape Streamer for above £550.00

- Plug in Cards
- Hardware
- Software
- Disk Drives

DATA SHEETS AVAILABLE ON MOST PRODUCTS

For more information:

Peanut Computer
Freepost
Dewsbury
WF13 1BR

(No stamp required)
Normal Delivery:
Low Mill, Dewsbury WF13 3LX
Telecom Gold 72: DTB 10199

DINERS CLUB INTERNATIONAL · AMERICAN EXPRESS · VISA

Computer

Phone 0924 499366
(24 hour answering service)

Money back guarantee. Add 15% VAT to all prices. Carriage (for orders under £100) – £3.00 (for orders over £100) – £5.00. Monitors £7.00 – Printers £7.00. Prices may fluctuate with £ rate, goods will be charged at prices ruling at date of despatch.

We will match the price of any item elsewhere, provided documentary proof is given, and the item is ex-stock. Dealer enquiries welcome inc. IBM.

Peanut is the trading title of Golf Computers Ltd.

Lap Mac on way?

Strong rumours of even more new Apple products on the way are circulating the industry.

From America comes news of a portable Macintosh. Word has it that the laptop will have LCD screen and disc drive and could sell at around £350.

Before then, a new multi-tasking operating system for the Macintosh II is expected on the market.

Under the code name Juggler the system has apparently been shown to software developers and could become available before the end of the year.

Apple to produce its own software

APPLE Inc is setting up its own software development company.

Though initially a wholly-owned subsidiary publishing under its own label, the new company is expected to become independent within a year.

It will develop, publish and market applications software for the Apple II and Macintosh, including applications software currently published by Apple. But it will also publish new products from other third parties.

Apple chief executive John Sculley said: "We believe the creation of this company will expand and strengthen the software industry created around our products".

He said that since Apple was putting less emphasis on its logo and label in the software field, developers could participate on a more equal footing.

"The new company will increase the opportunities for third parties to bring important new software applications to the market.

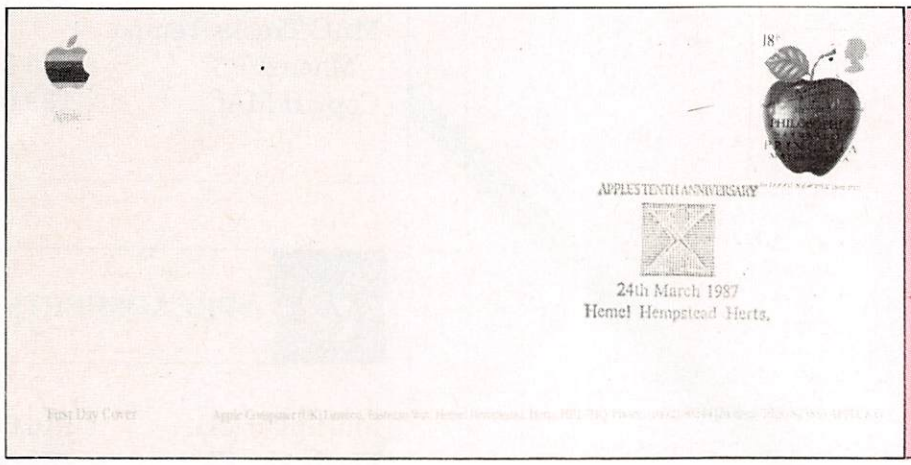
"Right now, there are many small developers working on breakthrough software for Apple products, but they don't have the resources to market and distribute their products. The new software company will provide them with a significant opportunity to do so.

"It will ensure that a full and complete line of quality applications software is available for Apple products.

"By focusing primarily on Apple, it can easily target untapped or emerging markets", said Scully.

Running the new company to start with will be William Campbell, formerly Apple's executive vice president of US sales and marketing.

When the company eventually goes independent, Apple will retain a minority stake.



STAMP OF APPROVAL

AS part of its tenth anniversary celebrations, Apple UK sponsored a limited series of first day covers. The company considered it particularly appropriate that the stamps used were those marking the 250th anniversary of the death of scientific pioneer Isaac Newton and that the 18p stamp featured a symbol shared by both of them – an apple.

Battling on Apple II

LATEST release from The Avalon Hill Game Company is a strategy game for the Apple II.

Darkhorn is set in the future, where four major powers are battling for supremacy over land belonging to the Darkhorn mountain.

Players enter villages, foothills and woods to recruit supporters and eventually overthrow the Darklord.

Up to four players can take part and a total of 16 different locations are featured. Price $30.

SALES ARE SOARING

STRONG gains by both its Apple II and Macintosh product families have helped Apple Computer to a 41 per cent increase in net sales.

New products also contributed substantially to the worldwide growth in revenues, the company said.

Sales for the second fiscal quarter of this year were £353.5 million compared to £251.2 million for the same period last year.

Net income for the quarter was £21 million, compared to £19.6 million a year ago.

Net income increased less than revenues due to an increase in research and development spending "to fund extensive new product development and a planned reduction in gross margin".

Research and development expenses rose 43 per cent from the previous year as Apple continued to launch major new products.

"Results for the quarter are much better than we had expected", said Apple chief executive John Sculley. "Sales momentum has continued to build in both US and international markets".

David Hancock, Apple UK managing director, said: "We're particularly pleased with the initial acceptance of the Macintosh SE.

"Initial orders are the strongest for any new Macintosh product in Apple's history".

THIS MONTH'S SUPER SAVERS

WORD PROCESSING

Mac Publisher	£135.00
Aldus Pagemaker	£395.00
Format-80 (II)	£99.00
MAC Author	£159.00
MAC Word	£120.00

PRINTERS

Epson LX 86	£210.00
Epson FX 85	£339.00
Epson FX 105	£425.00
Epson JX 80	£375.00
Epson LQ 800	£460.00
Epson LQ 1000	£585.00
Epson FX 800	£325.00
Epson FX 1000	£435.00

GAMES SOFTWARE

Flight Simulator II	£29.00
Flight Simulator Mac	£35.00
Mac Checkers & Reversi	£23.00

FORMAT 80 SCIENTIFIC £139

List price £179
"as reviewed in January Apple User"
Greek alphabet, and scientific symbols. What you see is what you get. Price includes telephone support. One day training courses also available

GRAPPLER + £65

List price £99
Offers dual hi-res graphics, allowing you to dump page one and two simultaneously. Can also dump text and graphics simultaneously. Images may be enlarged, inversed, or rotated through 90°.

A complete catalogue is available with details of

PRINTERS & ACCESSORIES
PLOTTERS & ACCESSORIES
MODEMS
APPLE II & MAC HARDWARE
APPLE II & MAC SOFTWARE

ALSO AVAILABLE:

EX-DEMO STOCK AT UNBEATABLE PRICES

DATABASES

Reflex Mac	£80.00
dBase II (CP/M)+Z80	£255.00
Omnis 3 MAC	£325.00
MAC File	£120.00
PFS: File & Report MAC	£39.00

UTILITY PROGRAMS

Autoworks	£45.00
Copy II Plus (II)	£39.00
Sideways Apple (II)	£40.00
MAC Memory Disk	£28.90
Work & Print Spooler MAC	£24.65
MAC Tracks Tempo Macros	£65.00
Copy II MAC	£39.00

SPREADSHEETS

Multiplan (II)	£76.00
Flashcalc (II)	£69.00
MAC Multiplan	£119.00

Please add £5 for P&P on printers, £1.50 on software and 15% VAT to total.

RCS Ltd
132 Evelyn Crescent
Sunbury-on-Thames
Middlesex TW16 6NA

0932 761815

Access

Mail Order Only

'Too realistic' simulation is banned

AN award-winning computer simulation that's a best-seller on the Apple 11 has been banned in Germany for being "too realistic".

Silent Service – an accurate portrayal of a World War II submarine in action – can now only be bought from regulated outlets such as sex shops in the federal republic.

The decision to outlaw the title, called Das U Boot in Germany, has been taken under legislation known as the Youth Dangerous Publications List.

A controversial law passed to protect German youngsters, it covers a range of products from pornography through to material thought likely to incite aggressive behaviour.

This is in fact the second time that publisher MicroProse (0666 54326), the world leader in entertainment simulations, has fallen foul of the List.

The first of the giant software house's titles to be banned in Germany was F-15 Strike Eagle, also a chart topper on Apple machines. One of the best known entertainment simulation packages, for the Apple, the combat flight simulator was in the American Billboard Top 20 for more than a year.

While admitting both titles are accurate in all details, MicroProse denies suggestions that either F-15 Strike Eagle or Silent Service is likely to encourage anti-social behaviour among German youth.

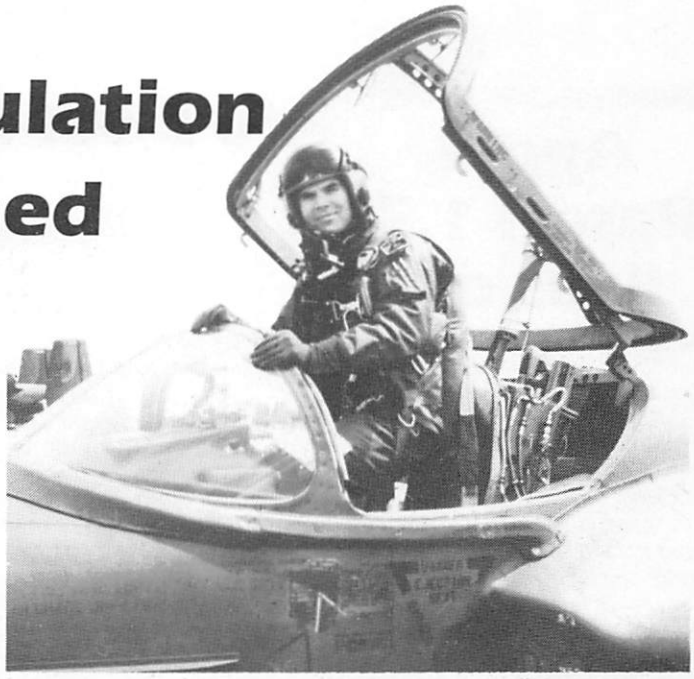
The second ban has been sufficient to goad MicroProse into lodging an appeal with the German authorities.

In fact, company president Bill Stealey – a fighter pilot who is an adviser to the US Joint Chiefs of Staff – flew over to Munich recently to discuss the situation with local legal experts.

"We believe the time has come to take a stand", he told *Apple User*.

"For this legislation – wrongly in our opinion – lumps together computer software with pornographic videos.

"To drive sophisticated software into the back streets is not only harmful to companies like ours, but will have a damaging effect on Germany's own burgeoning software industry".



MicroProse president Bill Stealey: "Time has come to take a stand"

More for the Macs

HIGH level support for the new Macintosh SE and II continues to flow on to the market.

The Nantucket Organisation expects to have its new database package for the machines out in July.

The McMax, which is dBase compatible, will cost £295 and, says Nantucket (0992 554621) allows users up to 2,000 fields per database, 2,000 memory variables and 2,000 procedures per file.

Lotus is working on a modular integrated business package which it hopes to have on the British market by the end of the year.

Using the working title Galaxy – not necessarily the final name – it will have graphics, word processing, worksheets, forms and communications modules plus its own language.

Users will be able to transfer Lotus 1-2-3 and Symphony files to and from Galaxy.

No price has been set for the product as yet says Lotus (0753 840281).

Moves at the top

APPLE Computer has announced changes in its managerial ranks "to broaden the breadth and depth of executive leadership".

Chief executive officer John Sculley has named David Barram to a new post as vice president of corporate affairs.

Replacing Barram as vice president and chief financial officer is Deborah Coleman, formerly vice president of worldwide operations.

Her position is filled by Ralph Russo, currently director of international operations.

MOUSE HAS A WAY WITH WORDS

A NEW mouse-based word processor for the Apple II series has been released by Sensible Software.

Sensible Writer is designed for production of letters, reports, essays and other home and business writing.

It features a Macintosh style interface and can be used with a mouse, keyboard or both.

Features include pull-down menus, built-in mail merge for personalised form letters, rulers to format documents, and automatic envelope addressing. It also has the ability to handle two large documents simultaneously.

The user can read and write AppleWorks and text files directly. The program automatically uses extended memory cards, utilising up to 512k from them, and it loads completely into memory and installs on hard discs.

Sensible Writer is suplied on 5.25in and 3.5in discs for the Apple IIGS, IIc and enhanced Apple 128k IIe. Price $99.95 from Sensible Software in the USA (0101 313 258 5566).

Digitiser for all Apple IIs

AN enhanced video digitiser for all Apple IIs has been produced by Stem.

ComputerEyes 2 produces a high resolution digitised picture from camera, video recorder or any other video source.

Stem (0382 65113) says image capture in all modes takes less than six seconds.

Images are displayed on the Apple screen and can be saved to disc for recall.

The product comprises a single board peripheral slot card with full menu-driven software on disc. This can acquire pictures in various modes, including grey scale simulation and double hi-res, and adjust contrast and brightness.

True grey scale image can be obtained with the Video-7 enhancer or the Legend 'E Card.

An optional enhancement disc (at £29) allows direct capture of Print Shop Graphics, Newsroom Photos and various format changing of images – useful in desktop art applications – plus image negation, expansion, contraction and scrolling.

Apple II Plus and IIe versions of the product cost £129; upgrades for the IIe to IIGS cost £39; and ComputerEyes 2 source code is £40.

Apple Pascal 1.3 lives on

APPLE has moved to stamp out rumours that the days of Apple Pascal 1.3 are numbered.

"Apple has made no such announcement in either the UK or the USA", said Apple II product manager Clive Girling.

"Apple Pascal 1.3 is a living product and will continue to be so for some time to come.

"While it is true to say that most development work is likely to switch to the new 16 bit Apple Programmers Workshop running on the IIGS – which includes 65816 Assembler, C and Pascal – we have no intention of restricting developers to this environment.

"In fact in those countries where the IIe continues to be available it may well be the environment of choice for those developers working on IIe specific products".

Art on Mac Plus

THREE products for the new Macintosh Plus have been introduced to the UK market by distributors Heyden and Son.

MacScan is a high-speed interface for the Canon IX12 tabletop scanner which can quickly capture line art and half tone images.

A program available with the interface allows the editing, saving and printing of scanned images and supports major desktop publishing programs' image save formats. Price £1,895.

Memorandum users can tag memos electronically onto Mac spreadsheets. The program allows the addition of text or graphics without any modification to the original documents. Price £99.

Voila! is a desk accessory outliner with which any word processing document can be transformed into an outline in seconds, automatically.

There is no limit to the number of headlines or subheads users can create. It is possible to edit fonts, sizes and styles for a single headline, a level, a whole family or the whole outline.

Four formats are available: Arabic, Roman, Standard or Symbol. For each outline a table of contents is instantly created. Price £99.

NEW FLIGHTS OF FANCY

A NEW scenery disc for Microsoft Flight Simulator on the Macintosh has been released by SubLogic Corporation.

It covers details of the East coast of the United States from Washington to the tip of Florida.

Featured are many rivers and roads, railroads, racetracks, transmitter towers – some with blinking lights – and elevated bridges which cast shadows.

Detailed "sightseeing" areas are available, along with more generic scenery areas which include radio navigation aids.

More than 130 airports, a dozen of them military, are incorporated. Price $24.95 from SubLogic (0101 800 637 4983).

BT titles collect 31 awards

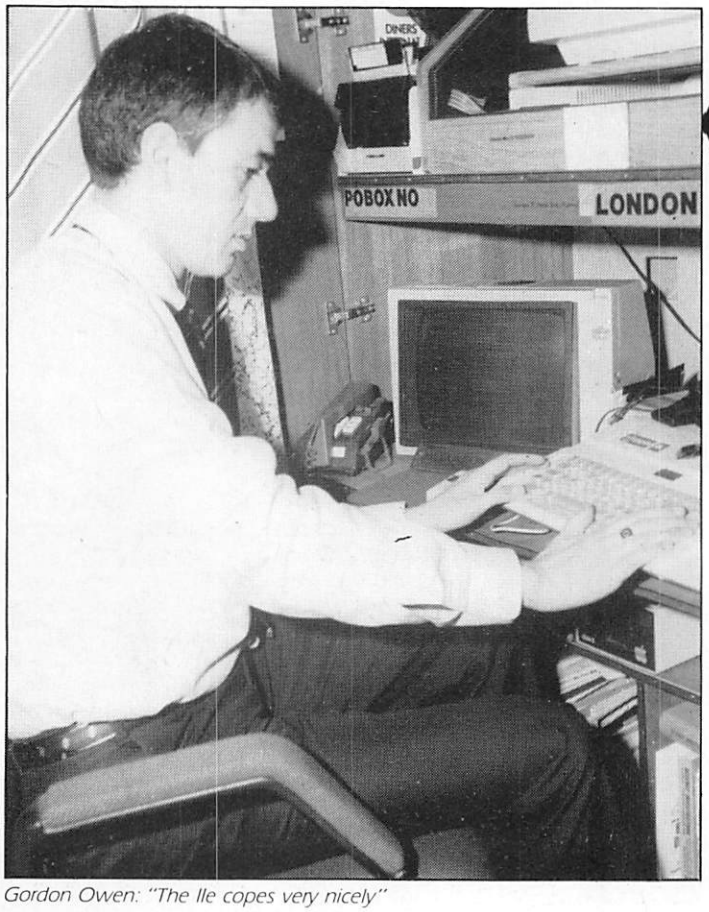
BRITISH Telecom's software titles have scooped an amazing total of 31 awards here and abroad in the past 12 months.

The accolades were awarded by reader polls and computer journalists' votes in magazines published in Britain, France, Germany and the United States.

More than two-thirds of Telecomsoft's sales of programs under the Rainbird, Firebird and Beyond labels are now overseas.

Latest release from Rainbird (01-240 8838) for the Apple II and Macintosh is Guild of Thieves. Guild is a new adventure from Magnetic Scrolls to follow the success of The Pawn which was voted Adventure of the Year at the Golden Joystick Awards.

It is set in the mystical land of Kerovnia, familiar to players of The Pawn, and features a large number of complex puzzles. Price £19.95 for the Apple II and £24.95 for the Macintosh.



Gordon Owen: "The IIe copes very nicely"

Have a party on the Apple

THROWING a party in Greater London or Essex is a lot less onerous thanks to Gordon Owen and his Apple IIe.

For the past six years he has been operating a computerised Rent-A-Bar service to help the hosts and hostesses of all types of functions.

The firm provides bar equipment from coolers and optics to ice buckets and cocktail sticks for small family celebrations and large civic functions alike.

More importantly, it also supplies the essential ingredients for encouraging sociability – beer, wine, spirits and, increasingly these days, non-alcoholic drinks.

The Apple IIe helps calculate what each function needs, then prepares lists of requirements to be sent to breweries and other suppliers, and keeps track of the stocks and orders.

"Apart from taking a lot of the hassle out of organising a function, we also help our customers save money", said Gordon Owen.

"We price each function individually and shop around very carefully for the best priced drinks before we offer a quotation.

"This means we have to contact several firms each time, and coordinate deliveries from different sources.

"With the help of dBasell, the Apple IIe copes very nicely.

"Our main problem area is in communications because ordinary mail is very slow and a bad telephone connection could lead to expensive mistakes.

"I wish more of the people we deal with would follow our example and join MicroLink and Prestel because electronic mail is the perfect solution".

Software surgery – for fun and profit

The first in a series of articles in which Duncan Langford helps you make the most of your Mac

YOUR own, personalised desktop? The ability to shut your Mac down without using the mouse? It really can be quite easy to achieve...

Anyone who has used a Mac for a while builds up their own library of useful programs, but in practice most people limit the ones they actually use to a mere handful.

Good Mac programs do seem to cost large amounts of money. If your basic processing needs are covered, there's perhaps no real incentive to buy more; after all, dealers are understandably reluctant to allow much testing before purchase, and the new program may not live up to its advertising.

ShareWare is usually cheap, but documentation is notoriously poor. The same criticism applies to the applications for programmers released by Apple, such as ResEdit. Most ordinary users just don't have the time to puzzle out instructions for themselves.

This is a pity, because some of the available programs are very good indeed, and well worth setting beside commercial software. Indeed, some effects are possible with this "non-commercial" software which just aren't practical with off-the-shelf applications. What is needed is an easy to follow introduction to selected non-commercial software – which is what I hope to provide, in this and subsequent articles.

I'll begin by giving an introduction to a very useful – but rather intimidating – Mac programmers application: ResEdit. This is an application which, among other things, will allow us to modify the icons on the Mac's desktop – and to change menu items. In fact, once you discover just how easy it is to modify your Mac, your desktop may never be the same again.

To start, you will need a copy of ResEdit. I used ResEdit 1.0, of August, 1986. It's widely available from user groups, MacTel, and most dealers.

You'll also need a disc containing just the System and Finder. Copy your ResEdit 1.0 on to this disc (always work from a copy) and, as we're going to modify the Finder, you'll need to copy this, too.

Just as an exercise to begin, we'll change the description of the Wastebasket (which in this American version of Finder 5.3 is called Trash) both where it appears in the normal desktop Special menu bar, and in the wording under the dustbin icon. Finally, to simplify the process of closing our Mac down, we will allow Command-Q to act as the equivalent of selecting Shut Down from the Special menu.

Figure III shows what the menu looks like now.

Details of these items are kept in the Finder, so that's where the changes must be made. After making a copy of the Finder, select it and choose Copy from the Edit menu.

We start by double-clicking on the ResEdit icon. The disc will hum for a few moments, and you will see a screen similar to that in Figure IV. My disc was called Spare Disk – yours will probably have a different name.

Note that all files are shown, even those which are normally invisible. An example is the DeskTop file, which, among other things, remembers the position of desktop icons.

As we're interested in the Copy of Finder file, double-click on that. Most Mac programs and applications are constructed a little like nested folders; think of our Finder as being a large folder containing other folders.

ResEdit allows us to open up this main folder, and then to open further ones in order to locate the part of the application we're looking for – the bit which deals with the Finder menu bar. Helpfully, this is located in a folder called Menu (see Figure V).

Other areas inside this folder contain

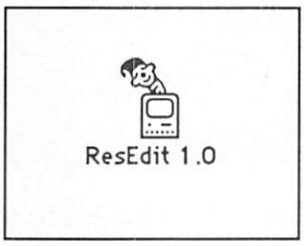


Figure I: ResEdit 1.0 icon

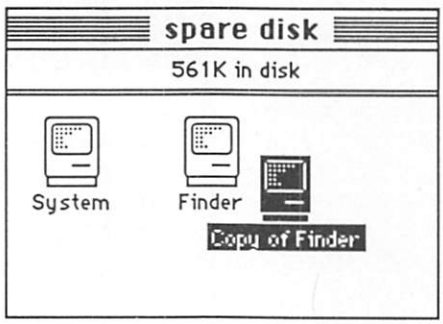


Figure II: Always work from a copy

Figure III: Original menu

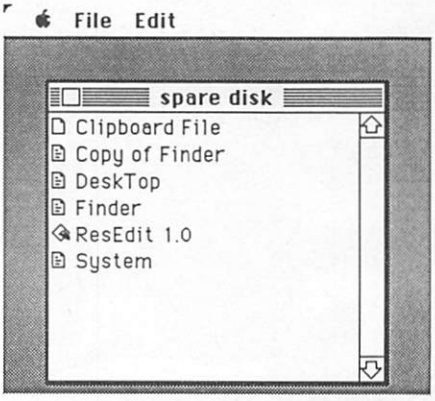


Figure IV: The first ResEdit window

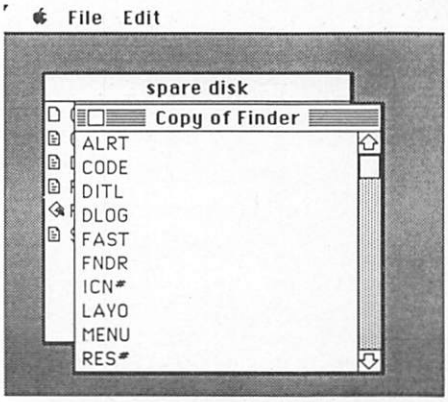


Figure V: Inside the main Finder window

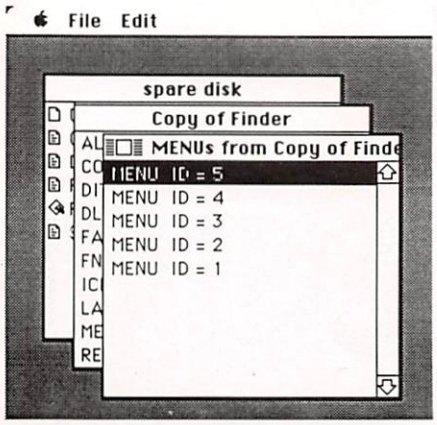


Figure VI: Menu window open

icons, dialogue boxes, the program code; in fact, everything which makes up the Finder. As you'd expect, to open up the Menu, double-click on the word Menu.

There are of course five menu headings (File, Edit, View, Special and the Apple menu) so there are naturally five items within this window (see Figure VI):

The Special menu is the fifth, counting from the left of the Mac screen, so to look inside it we double-click on Menu ID = 5. This will open up the final window, a long one containing all the information concerning this menu item. The title Special is visible in a box at the top of the window (see Figure VII) and, as you scroll downwards, other elements become visible.

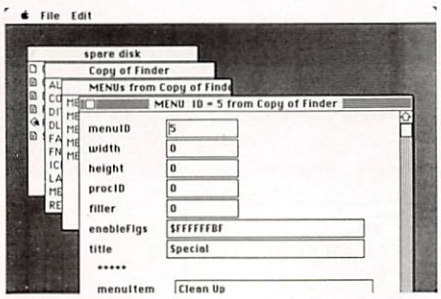


Figure VII: Menu ID = 5 open

Each option within the menu has its own separate section in this window; scroll it until you see a box containing the words Empty Trash (see Figure VIII).

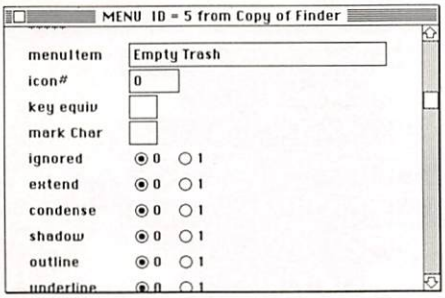


Figure VIII: Empty Trash

We're ready to make the first change. Click inside the box, and then change Empty Trash to Empty Wastebasket or to whatever you would like instead. That's it. Simple, wasn't it?

Scroll on a little further until you see the Shut Down menu item. Just under it is another box, titled key equiv, this time without any typing in it (see Figure IX). Click in this box, and type an upper case Q. You could type anything you liked, provided that it wasn't already used by another menu item as a command+key shortcut. Q is used generally to mean Quit, though, so we'll stick to that.

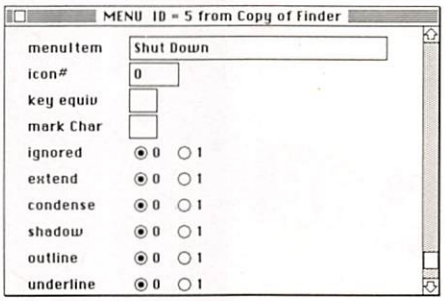


Figure IX: Shutdown

We've almost finished. So far, we have changed the description of the Trash in the Special menu to read Wastebasket and added a Control+Q command, to allow the Mac to be shut down from the keyboard. To save these changes, close all open windows. ResEdit will ask you whether you are sure you wish to make changes; click Yes.

We need to make one last change: The dustbin icon is still labelled Trash. Hmm. Back to ResEdit, but this time after opening our copy of Finder, ignore the Menu item. We want to change a text string; so STR# is the place to double-click (see Figure X).

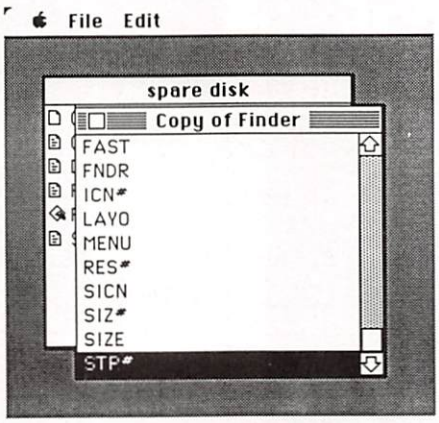


Figure X: Select the text window

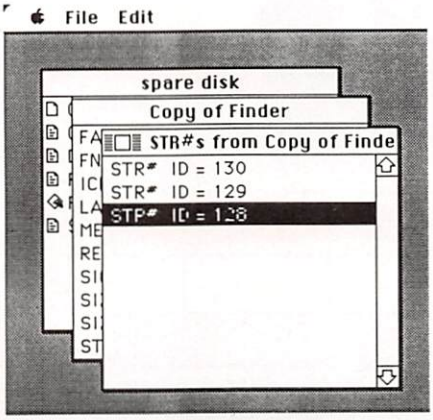
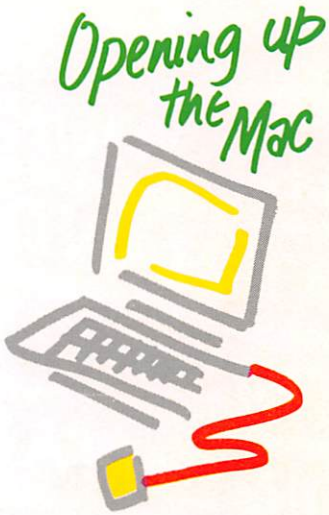


Figure XI: The Str# window



Inside this window is, as you'd expect, another; however there are only three further windows. Explore them all, if you want. The one we need though is the last one, STR# = 128 (see Figure XI).

When opened, you will see that this final window contains the names of many familiar objects. Almost at the top, though, you will see the string Trash (see Figure XI).

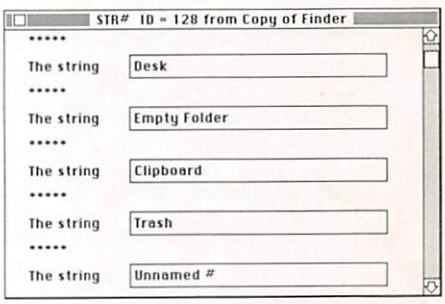


Figure XII: The str# 128 window

By now you should have a good idea what to do next. Click in the Trash window, and change the word to Wastebasket. And that really is all.

Close ResEdit, and move your modified Finder to a new disc. (I suggest you make a note in the About Finder window, to make it clear that this is a modified version).

Don't forget to rename it Finder: Copy of Finder won't work. Add a System to the disc, and close down the Mac. Now, when you boot your new Finder, you should see a Special menu which looks like Figure XIII:

Figure XIII: The changed menu

In future articles I hope to describe other ways of using ResEdit, to have a look at some of the programming desk accessories, and generally to show how readily available applications can totally – and easily – personalise your Mac. □

AppleUser SPECIAL OFFERS!

SAVE UP TO £10

TRY YOUR AT HAND THIS TOP ORIENTAL STRATEGY GAME

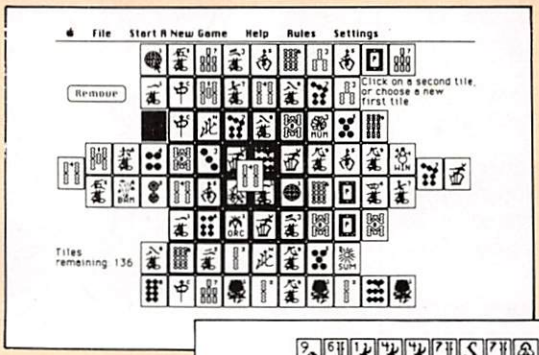
SHANGHAI is based around the 3,000-year-old Chinese game Mah Jongg which took the US by storm during prohibition in the 1920s – and promptly got banned when it turned many unsuspecting gamesters into ivory tile addicts.

The game consists of 144 tiles depicting flowers, seasons, dragons, the wind and more. These are stacked up to five tiles high in the Dragon formation.

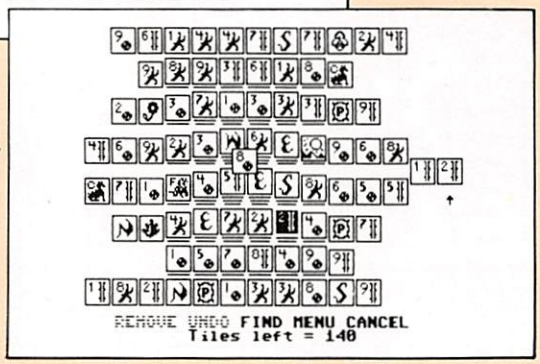
Your task is deceptively simple: Snap off matching pairs from the top of the piles until you run out of tiles.

You can play against the clock, in a team, or against your friends – as well as take back moves, peek under tiles and call up the Help screens.

This delightful, challenging game will keep you and your friends engrossed for hours.



Macintosh version



Apple II version

Suitable for	RRP	Special Reader offer	YOU SAVE	Offer including subscription	YOU SAVE
Apple IIe/Mac	£24.99	£19.99	£5	£32.99	£10

TO ORDER PLEASE USE THE FORM ON PAGE 73

AppleCentre Chiswick

At AppleCentre Chiswick, we have made it our business to keep up to date with new developments in Macintosh products. So if you are about to invest in improving your business efficiency, our experts will help you through the stages of decision making step by step. We offer the best advice, training, maintenance and support.

Desktop Publishing

PageMaker

Finance

Microsoft Excel

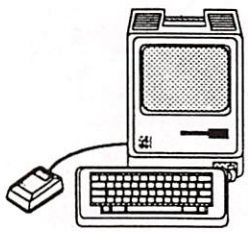
Information

Omnis

Word Processing

Microsoft Word 3.0

Macintosh Family



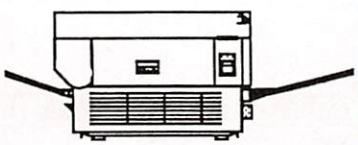
Macintosh II
Available now

Apple II GS

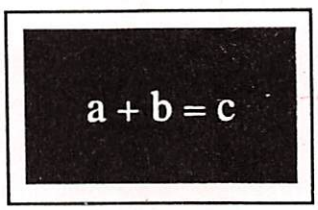
IIGS

Aplle IIe
Ex-demo for sale

Bureau Services



Training



Rentals

AppleCentre Chiswick 118 Chiswick High Road London W4 1PU Tel: (01) 994 7424
Tasha Business Systems 191 Kensington High St London W8 6SH Tel: (01) 937 3366

Authorised Dealer

Logo and the Mac

Duncan Langford tries out ObjectLogo and enjoys the experience

LOGO is the name of a programming language for the Mac. But wait – don't assume that programming languages are only of interest to hackers. Logo is different.

Let me digress for a moment. As a teenager, I was interested in motorbikes. My crowd used to think that there were two types of motorcyclists – those who used their machines, not apparently caring about how and why they worked; and those who enjoyed taking bikes apart so much that they never actually seemed to get them out of the garage. There may perhaps be a similar way of looking at Macintosh users.

The first group of users assume their computer consumes only packaged food, and therefore provide it with an exclusive diet of (expensive) proprietorial software. The other group looks on the Macintosh as a challenge; they never buy software, and spend their time happily hacking away into the system.

Unlike those teenage mechanics, though, Mac programmers really do use a different language from the rest of us – a programming language – and they use it to communicate with the Mac.

What is a programming language? Think of it as just a way of telling the computer what you want it to do. Programming languages come in various levels – normally, the higher the level, the easier a computer language is to understand, but the less flexible it tends to be. (Real programmers tend not to use high level languages at all, preferring something called assembler – but that's another article).

There are many, many different languages available for micros, each designed at different times from different perceptions of need and function. Some are relics of old main-frame computers, while others are of very recent design.

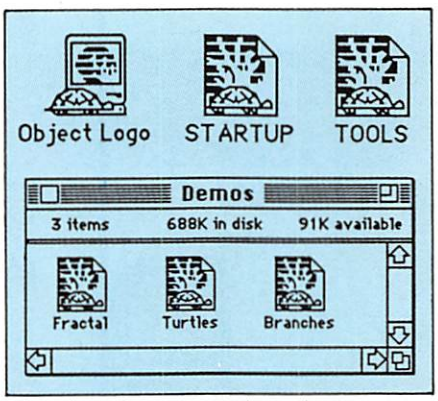
The Mac has an enormous number of these programming languages available – far more than most computers. This is perhaps because the Mac (which comes with no "packaged" programming language) broke from a tradition going all the way back to 1977, when the first Commodore PET came with the language Basic built-in, stunning reviewers accustomed to loading computer languages from cassette tape.

Back to Logo: Although computer languages understandably tend to be designed for computer programmers, Logo is totally different. Logo was designed for non-programmers.

Skip this paragraph if you're a believer in the exclusivity of computers, because the essence of Logo is ease of access – it's probably safe to tell you, now you've read this far, that originally the language was designed to make computers accessible to children.

Seymour Papert, developer of Logo, was a mathematician who felt that mathematics as conventionally taught was inevitably boring. By opening up the graphics abilities of computers, through Logo, even very young children could learn while actually enjoying themselves.

Think of Logo as being very like Lego – those little children's bricks which clip together. You don't buy a completed Lego toy; you build one, from a small range of



ObjectLogo icons – note the turtle

bricks. When it's finished, you can pull it apart, and build something else, using the same bricks.

Logo is like that. It has a small range of instructions, or primitives. These may be combined with each other, to make more complex instructions. These more complex instructions, in turn . . . yes, you've got the idea.

Central to the philosophy of Logo is the concept of programmable objects, principally a turtle – yes, turtle – having the ability to draw lines. Turtle graphics follow the concept of a programmable turtle, which may be instructed to draw whatever shapes you wish. Complex pictures may be made up of combinations of simple pictures, in exactly the same way that the most complicated building may have been designed to be constructed of ordinary bricks.

As an example of the Turtle in action, let's draw a simple box on the Mac screen. Don't worry, incidentally, about the triangular shape you can see when drawing Logo graphics; this is the Logo turtle, and it can easily be hidden with the command, HideTurtle, or HT.

The normal Macintosh screen is made up of single dots, or pixels: ObjectLogo allows us to use these as units of measurement. So, to draw a box with sides measuring 50 pixels, the Logo instructions to our turtle would be:

```
Forward 50
Right 90
Forward 50
Right 90
Forward 50
Right 90
Forward 50
```

Effective and easy to follow, but lengthy, although commands can be shortened – Forward to FD, for example.

Fortunately, Logo allows procedures to be built up, combining simple instructions into more complex ones. To draw our square, we could first define a side, of changeable size. (We can allow different sizes by providing a variable name, say *size*, after a colon; it isn't essential).

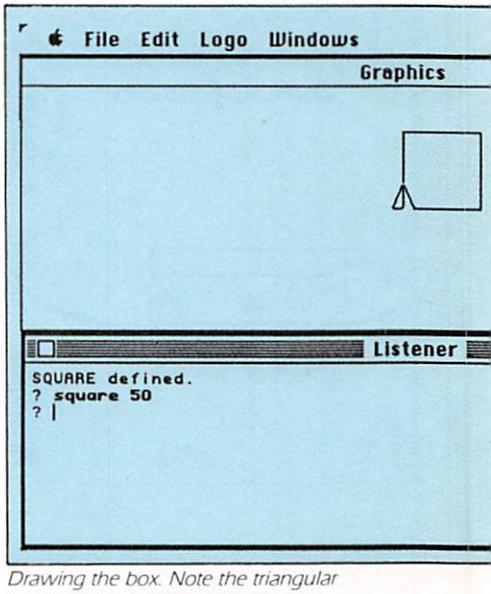
The first word typed, To, tells Logo that we don't want these instructions to be immediately carried out, but instead intend defining a Logo procedure. Our procedure will be known to Logo by the name immediately following the To instruction – in this case, side.

```
To side :size
Forward :size
Right 90
End
```

And to make a further procedure, which will use our newly defined procedure *side* to draw the square:

```
To square :size
Repeat 4 [side :size]
End
```

Finally, to actually draw the square, a



Drawing the box. Note the triangular turtle – the command HideTurtle, or HT, will make it invisible

simple instruction is typed. The instruction? Well, for a square with sides of 50 pixels:

```
square 50
```

And that's all. For a square with sides of 100 pixels, we just change 50 to 100. The power of the language Logo is illustrated by the fact that our newly defined instruction can be used repeatedly, on its own or combined with others, exactly as if it had always been built-in to the language.

For people expecting programming a computer – particularly a Macintosh computer – to be complicated, Logo is therefore a revelation.

There have been several offerings of Logo for the Mac, but the latest, ObjectLogo, is much the most powerful. It not only has comprehensive turtle graphics, which even allow multiple turtles, but also permits full access to the deeper workings of the Mac.

This will be of much interest to professional programmers, permitting as it does full Toolbox access, multiple windows, menus, full I/O access... in fact, the range of likely users of ObjectLogo is enormous. After all, there can't be many programming languages ideal for beginners to programming, but which hackers are likely to find useful.

The program itself comes on a single 800k disc, together with supplementary programs and examples. I particularly liked Coral Software's new font, which allows both bold typing by the operator and standard printing for the Mac's replies –it's called Schizo!

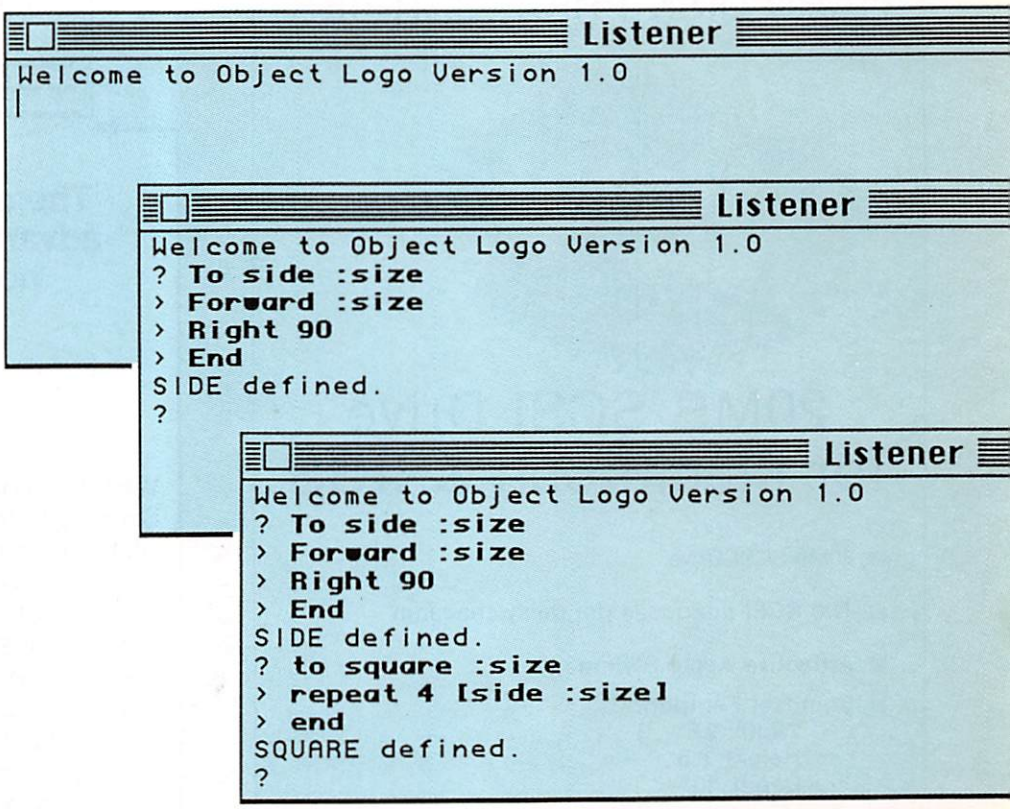
With the programs comes a manual which measures nine inches by seven and a half – and is more than an inch thick. As well as very comprehensive descriptions, the manual contains many printed program examples, which are also provided on disc – a very useful way to develop knowledge of how a language works.

Although the manual is impressively comprehensive, Coral Software suggests that beginners to Logo first study a simple book on the language; I feel this is good advice. The manual itself probably holds all the information you are likely to need, as well as much that you won't; it consequently may well appear rather intimidating to a new user.

The implementation allows the same simple approach we used to draw a square, using a turtle object, to be followed in non-graphic applications, with very real benefits in ease of understanding and power. In fact, object-oriented languages – Lisp is perhaps the best known – are used in artificial intelligence research partly for these reasons.

Overall, I was very impressed by ObjectLogo. A full interpretation of the language Logo, it has a particularly powerful link into the Macintosh operating system, placing it on a par with other high-level languages such as Pascal, or the new Microsoft Basic – but at a much lower price.

Despite the power of ObjectLogo, it would be an ideal first programming language for someone new to computing,



The ObjectLogo Listener window, where all commands are typed – in this case defining the procedure side and square

and of course, as it is based upon an object-oriented approach, it is conceptually easy to follow.

At a price of only around £60 ObjectLogo has to be the best bargain in programming languages for some time. Highly recommended.

Math functions: Described by Coral Software as the most advanced mathematics ever put on a micro "unbounded integers, complex and rational numbers, and a full set of transcendental functions."

Toolbox: Full access, with low-level access to all Macintosh rom. Mac data structures, such as windows and menus, are implemented as ObjectLogo objects, making high-level access easy.

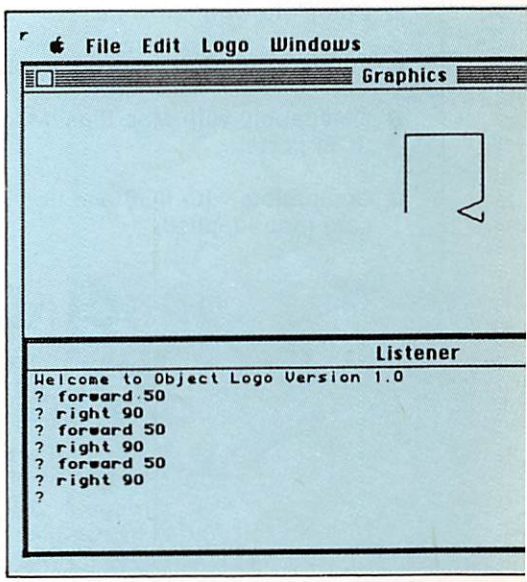
Debugger: The built-in debugger provides full step, trace and watch commands.

Compiler: An incremental compiler, generating native 68000 in-line code from each line and procedure. Consequently, very fast running programs.

Windows: Six customisable classes of windows, again implemented as ObjectLogo objects : generic windows, turtle windows, text windows, file windows, editor windows and listener windows.

Listener: (The path for your instructions to reach ObjectLogo). The Listener window is fully scrollable, allowing reference back to all earlier interactions.

Input/Output: All I/O is implemented by object oriented streams, including random access files and window and serial I/O.



Drawing the square in immediate mode

Where to buy ObjectLogo

ObjectLogo is available by mail order from the developers, Coral Software, PO Box 307 Cambridge, MA 02142, USA at a price of $89.95, to customers in the UK.

However, I found the easiest way to order was by telephoning with an Access or Barclaycard order (in the late afternoon – remember the time difference) on 01-01-617-868-7440.

Despite the extraordinarily low price, Coral offer an unconditional 60-day money back guarantee, which really can't be bad.

□

BIDMUTHIN SMASHES THE HARD DISK PRICE BARRIER

NEW

20MB SCSI Drive for Macintosh & IIGS

- 20Mb SCSI Drive
- Two SCSI interfaces (for daisy-chaining)
- Attractive Apple Styling
- Compact Footprint
  Width: 2.5″
  Height: 6.5″
  Depth: 10″
- Plug and Play — Pre-formatted
- Complete with software
- Internal 220V-50Hz power supply
- Compatible with Mac Plus, Mac SE, Mac II via SCSI port
- Compatible with IIGS and IIe via Apple SCSI card (not supplied)

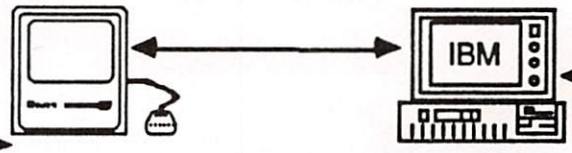
SPECIAL INTRODUCTORY OFFER
£499.00
+ carriage & VAT
With full 10 day money-back guarantee
Specify whether for Mac or GS.

Available only from:

BIDMUTHIN technologies

P.O. Box 264, Harrow, Middlesex, HA3 9AY
Tel: 01-907 8516
Telex: 8950511 ONEONE G
(Ref: 22554001)

TOPS



The computer industry's most advanced local area network is now available in the UK.

Mac to Mac: IBM to IBM
Mac to IBM to Mac
TOPS to UNIX

What is a Local Area Network (LAN)?
Simply, a LAN links previously isolated computers into what are commonly called workgroups. Workgroups can share files, hard disk storage and printers and can communicate with each other.

Why is TOPS so advanced?
Because it allows each computer to make available to anyone else on the network files that are in a local floppy or hard disk drive.
Because it is transparent. It works entirely in the background and does not disrupt the normal use of your computer even if a remote computer is accessing your files.
Because it can turn a dedicated hard disk into a network file server without any hard partitioning.
Remote users can run system and applications from HFS folders which have been 'published.' If you want to add a new user, create a new HFS folder. An HFS folder expands as you add data, so you will not run out of partition space like other programs.
Because it supports different operating systems. You can share spreadsheet, database and word processing files between the Mac and MS-Dos. You can share the LaserWriter. You can share hard disk space without partitions. How about an inexpensive hard disk PC clone as a very effective file server to a LAN of Macs!
TOPS works over Appletalk or PhoneNET.
TOPS is networking you can buy over the counter and set up yourself without any complicated configuring.
TOPS is revolutionising the office environment. Ring us now for details or your nearest TOPS dealer.
TOPS/IBM incl. Appletalk card £425.00+VAT.
TOPS for the Mac £149+VAT.
PhoneNet approx £50 per node.
All prices retail.

Go-Mark Ltd

Exclusive distributors for TOPS

3/10 Peter's Lane, London EC1M 6DS
Tel: 01-251 1160/1771

Basic lister offers easier programming

FROM the author of the very popular hi-res graphics editor (*Apple User*, June 1986) comes another very useful machine code program. It will be particularly useful to Apple II+ owners with 40 column screens, but is still worth typing in even if you have a IIgs, IIe or IIc.

Once installed in memory you only have to call it (you could set up the & vector to point to it) and it will list your resident Basic program on the screen in a particularly neat and easy-to-follow form. In addition you can stop and start the listing by pressing the spacebar and cancel it by pressing return.

Logic can be followed more easily than with a conventional listing because FOR ...NEXT loops are indented and sub-statements (those isolated by colons) are each begun on a new line. Also each line begins in the same place, which in itself is incredibly useful. As a bonus, control characters appear in inverse.

Paul Sinnett's machine code program makes on-screen listings simpler to debug

If you have an assembler, enter the text of the program and compile it. If you don't have an assembler don't worry: Call the monitor with the command CALL-151 and enter 6000:, followed by the machine code numbers given on each line after the colons. The first couple are A9 and 9.

You can enter quite a few lines of such numbers without having to press Return. However, make sure that you put a space between each number. After pressing Return, that code will be entered into memory and you can enter more by giving the address, followed by a colon. The address is the number to the left of the colon in the listing.

When all is entered, check your entries carefully by using the memory dump method (two addresses separated by a period will give a dump of the bytes) or use the list command (memory address followed by 1) to list it in a mnemonic form which will be similar to the listing given here.

```
SOURCE FILE: P.S.LISTER
0000:            1 ********************************
0000:            2 *            LISTER            *
0000:            3 * THIS UTILITY LISTS A PROGRAM *
0000:            4 * IN A NICER, EASER TO DEBUG   *
0000:            5 * FORMAT.                      *
0000:            6 *                             *
0000:            7 * HERE ARE THE CHANGES :-      *
0000:            8 *                             *
0000:            9 * 1. EACH LINE BEGINS IN THE   *
0000:           10 *    SAME PLACE               *
0000:           11 * 2. FOR LOOPS ARE INDENTED    *
0000:           12 * 3. EACH STATEMENT IS ON A    *
0000:           13 *    NEW LINE                 *
0000:           14 * 4. PRESS ANY KEY TO HALT THE *
0000:           15 *    LISTING. PRESS RETURN TO  *
0000:           16 *    STOP IT.                 *
0000:           17 * 5. ANY CONTROL CHARACTERS IN *
0000:           18 *    THE PROGRAM ARE WRITEN IN *
0000:           19 *    INVERSE.                 *
0000:           20 *                             *
0000:           21 * WRITEN BY PAUL SINNETT      *
0000:           22 * USING THE TOOLKIT ED.ASM.    *
0000:           23 * COPYRIGHT (C) 1984           *
0000:           24 ********************************
0000:           25 *
0000:           26 * VARIABLES AND ROUTINES
0000:           27 *
ED24:           28 LINP    EQU  $ED24
FDED:           29 PRNT    EQU  $FDED
0024:           30 HTAB    EQU  $24
0085:           31 STAL    EQU  $85
0086:           32 STAH    EQU  $86
0087:           33 COML    EQU  $87
0088:           34 COMH    EQU  $88
0027:           35 WIDTH   EQU  $27
0009:           36 INDENT  EQU  $9
0000:           37 *
0000:           38 *
----- NEXT OBJECT FILE NAME IS P.S.LISTER.OBJ0
6000:           39         ORG  $6000
6000:A9 09      40         LDA  £INDENT
6002:8D 6C 61   41         STA  INCR
6005:A9 AD      42         LDA  £$AD
6007:85 00      43         STA  $00
6009:A9 D0      44         LDA  £$D0
600B:85 87      45         STA  COML
600D:85 88      46         STA  COMH
600F:A5 67      47         LDA  103
6011:85 85      48         STA  STAL
6013:A5 68      49         LDA  104
6015:85 86      50         STA  STAH
6017:A0 00      51 NEWL    LDY  £00
6019:A9 8D      52         LDA  £$8D
601B:20 ED FD   53         JSR  PRNT
601E:B1 85      54         LDA  ($85),Y
6020:8D 6D 61   55         STA  NXAL
6023:C8         56         INY
6024:B1 85      57         LDA  ($85),Y
6026:8D 6E 61   58         STA  NXAH
6029:D0 03      59         BNE  CONT
602B:4C 6B 61   60         JMP  END
602E:           61 CONT    EQU  *
602E:C8         62         INY
602F:B1 85      63         LDA  ($85),Y
6031:AA         64         TAX
6032:C8         65         INY
6033:B1 85      66         LDA  ($85),Y
6035:8C 70 61   67         STY  SPRE
6038:20 24 ED   68         JSR  LINP
603B:AC 70 61   69         LDY  SPRE
603E:AD 6C 61   70         LDA  INCR
```

PERIPHERAL CARDS

VISION

Colour Modulator + sound for IIe	£30
PAL for use with colour TV for II	£42
Apple 80 Column Text card	£19
RESOLUTION (80 col IIe)	£19
RESOLUTION 64 (80 col + 64K RAM) IIe	£29
RESOLUTION 128 (80 col + 129K RAM + DOS RAM disk) IIe	£79
MultiRam RGB 256K + AppleWorks expan. + DOS/ProDOS RAMdisk	£129
SCREEN 80 (80 col + softswitch) II	£44
Digitek SCREENMASTER 80 (80 Col + softswitch) II	£49
DMS C12 RGB for II or IIe	£45
DMS C30 RGB for II or IIe	£55

PRINTER CARDS

PARA-GRAPH printer card + cable	£30
Digitek PRINTMASTER III + cable	£45
Grappler + compatible + cable	£45
Orange Micro Grappler Bufferboard 16 – 64K	£29
CACHECARD 16 with cable	£50
CACHECARD 64 with cable	£60
Serial/Communication	£48
Serial Printer	£36

INTERFACING & CONTROL

IEEE-488	£79
Clock Card (battery back-up)	£59
Z80 In-Circuit Emulator	£199
6502 In-Circuit Emulator	£149
I/O Card (Four 8 bit ports + 2 counter/timers)	£59
IC Tester (Recognises & tests most 74 & 4000 series)	£99
EPROM Writer upto 27256 (EPROM socket on card)	£89
EPROM Writer upto 27256 Single gang (external pod)	£139
EPROM Writer upto 27256 Four gang (external pod)	£199
EPROM Writer upto 27256 Ten gang (external pod)	£299
Apple Graphics Tablet Interface (used, without manual)	£12
Apple ROM card for six ROMS (used, without manual)	£15

OTHERS

Z80 CP/M (II or IIe)	£29
Disk Controller	£28
16K RAM Card for II	£29
128K RAM Card for II	£89

DISK DRIVES

Apple II compatible drive	£92
Apple IIc compatible drive	£99
XEBEC Hard disks-Run DOS 3.3, ProDOS, CP/M and Pascal programs	
SIDER 1 (10MB with controller)	£649
SIDER 2 (20 MB with controller)	£795
BSIDER (Tape Back-up)	£599

PRINTERS

EPSON	LX-86	£219
	Tractor feed for above	£20
	FX 800	£339
	FX 1000	£454
STAR	NL 10 with parallel or Apple IIc interface	£239
	NX 15 with parallel interface	£339
	NL 10 Cut Sheet Feeder	£55

MONITORS

HANTAREX (Quality Italian monitors)	
12" Hi-Res (green screen)	£75
KAGA	
12" Hi-Res (green screen)	£109
Monitor Base (Tilt & Twist)	£12

ACCESSORIES

Numeric Keypad for IIe	£24
AC Cooling Fan	£24
Hitek superfan for II	£30
Two tone 5.25" disk box (20 disks)	£2.95
3.5" Lockable storage box (40 disks)	£8
Diskettes DS/DD Non-Branded (10)	£9

JOYSTICKS

Plastic cased, two button II or IIe	£19
Plastic cased, three button II or IIe	£22
Metal cased, two button II or IIe	£25
Arcade type joystick for II or IIe	£29

SOFTWARE

Format 80 Enhanced (DOS & ProDOS version) II or IIe	£99
AppleWorks 2.0 processor/spreadsheet/database	£140
Pinpoint 2.0 (AppleWorks expansion software)	£69
Multiscribe (Word processing with multiple fonts) IIe	£59
MultiRam CP/M RAM disk	£29
Visicalc Super Expander software	£19
Niceprint software for IBM PC	£20

DATA SWITCHES

PARALLEL and SERIAL Two way communication	
1 to 2	£49
1 to 3	£59
1 to 4	£69
Cross Over	£69

SPECIAL OFFER

A special word processing system offer with or without printer, consisting of the folllowing:
Apple //e 192K (used)
Monitor (used)
Disk drive
Disk controller card
128K extended 80 column card
Parallel printer card (serial version available)
Format 80 enhanced software

FEW LEFT ONLY £549

PRICE INCL. NEW STAR NL-10 PRINTER ONLY £779

DEALER HOTLINE
(021) 356 3828

All prices shown exclude VAT & delivery and are correct at time of press

DELIVERY:
£3.00 + VAT – orders under £100
£5.00 + VAT – orders over £100
£10.00 + VAT – printers, etc.


289 Birchfield Road,
Birmingham B20 3DD
Tel: (021) 356 7402
Telex: 334303 TXAGWMG

```
6041:85 24      71          STA  HTAB
6043:C8         72          INY
6044:B1 85      73          LDA  ($85),Y
6046:88         74          DEY
6047:29 80      75          AND  £$80
6049:F0 02      76          BEQ  LLOOP
604B:C6 24      77          DEC  HTAB
604D:C8         78 LLOOP    INY
604E:B1 85      79          LDA  ($85),Y
6050:F0 62      80          BEQ  NXTL
6052:29 80      81          AND  £$80
6054:F0 06      82          BEQ  CHAR
6056:20 E6 60   83          JSR  TOKEN
6059:4C 4D 60   84          JMP  LLOOP
605C:           85 CHAR     EQU  *
605C:A5 24      86          LDA  HTAB
605E:C9 27      87          CMP  £WIDTH
6060:90 0A      88          BCC  SCROL
6062:A9 8D      89          LDA  £$8D
6064:20 ED FD   90          JSR  PRNT
6067:AD 6C 61   91          LDA  INCR
606A:85 24      92          STA  HTAB
606C:           93 SCROL    EQU  *
606C:B1 85      94          LDA  ($85),Y
606E:C9 3A      95          CMP  £$3A
6070:D0 20      96          BNE  NOTC
6072:A9 8D      97          LDA  £$8D
6074:20 ED FD   98          JSR  PRNT
6077:AD 6C 61   99          LDA  INCR
607A:85 24     100          STA  HTAB
607C:C6 24     101          DEC  HTAB
607E:C6 24     102          DEC  HTAB
6080:A9 BA     103          LDA  £$BA
6082:20 ED FD  104          JSR  PRNT
6085:C8        105          INY
6086:B1 85     106          LDA  ($85),Y
6088:88        107          DEY
6089:29 80     108          AND  £$80
608B:D0 02     109          BNE  CCHR
608D:E6 24     110          INC  HTAB
608F:4C 4D 60  111 CCHR     JMP  LLOOP
6092:B1 85     112 NOTC     LDA  ($85),Y
6094:C9 1A     113          CMP  £$1A
6096:B0 12     114          BCS  NOCT
6098:A9 3F     115          LDA  £$3F
609A:85 32     116          STA  $32
609C:B1 85     117          LDA  ($85),Y
609E:09 C0     118          ORA  £$C0
60A0:20 ED FD  119          JSR  PRNT
60A3:A9 FF     120          LDA  £$FF
60A5:85 32     121          STA  $32
60A7:4C 4D 60  122          JMP  LLOOP
60AA:B1 85     123 NOCT     LDA  ($85),Y
60AC:09 80     124          ORA  £$80
60AE:20 ED FD  125          JSR  PRNT
60B1:4C 4D 60  126          JMP  LLOOP
60B4:AD 6E 61  127 NXTL     LDA  NXAH
60B7:85 86     128          STA  STAH
60B9:AD 6D 61  129          LDA  NXAL
60BC:85 85     130          STA  STAL
60BE:AD 00 C0  131          LDA  $C000
60C1:29 80     132          AND  £$80
60C3:D0 03     133          BNE  STOP
60C5:4C 17 60  134          JMP  NEWL
60C8:A9 00     135 STOP     LDA  £$00
60CA:8D 10 C0  136          STA  $C010
60CD:AD 00 C0  137 STLP     LDA  $C000
60D0:C9 8D     138          CMP  £$8D
60D2:D0 03     139          BNE  SCON
60D4:4C 6B 61  140          JMP  END
60D7:          141 SCON     EQU  *
60D7:29 80     142          AND  £$80
60D9:F0 F2     143          BEQ  STLP
60DB:A9 00     144          LDA  £$00
60DD:8D 10 C0  145          STA  $C010
60E0:4C 17 60  146          JMP  NEWL
60E3:4C 17 60  147          JMP  NEWL
60E6:          148 TOKEN    EQU  *
60E6:A5 24     149          LDA  HTAB
60E8:C9 20     150          CMP  £WIDTH-7
60EA:90 0C     151          BCC  NSCRL
60EC:A9 8D     152          LDA  £$8D
60EE:20 ED FD  153          JSR  PRNT
60F1:AD 6C 61  154          LDA  INCR
60F4:85 24     155          STA  HTAB
60F6:C6 24     156          DEC  HTAB
60F8:          157 NSCRL    EQU  *
60F8:E6 24     158          INC  HTAB
60FA:A9 80     159          LDA  £$80
60FC:8D 6F 61  160          STA  TOCT
60FF:AD 6F 61  161 LOOP1    LDA  TOCT
6102:D1 85     162          CMP  ($85),Y
6104:F0 29     163          BEQ  PRTO
6106:8C 70 61  164          STY  SPRE
6109:A0 00     165          LDY  £$00
610B:B1 87     166 LOOP2    LDA  ($87),Y
610D:29 80     167          AND  £$80
610F:D0 0C     168          BNE  NXTOK
6111:18        169          CLC
6112:E6 87     170          INC  COML
6114:A5 87     171          LDA  COML
6116:D0 F3     172          BNE  LOOP2
6118:E6 88     173          INC  COMH
611A:4C 0B 61  174          JMP  LOOP2
611D:18        175 NXTOK    CLC
611E:E6 87     176          INC  COML
6120:A5 87     177          LDA  COML
6122:D0 02     178          BNE  NINC
6124:E6 88     179          INC  COMH
6126:EE 6F 61  180 NINC     INC  TOCT
6129:AC 70 61  181          LDY  SPRE
612C:4C FF 60  182          JMP  LOOP1
612F:8C 70 61  183 PRTO     STY  SPRE
6132:A0 00     184          LDY  £$00
6134:B1 87     185 LOOP3    LDA  ($87),Y
6136:29 80     186          AND  £$80
6138:D0 0B     187          BNE  ENDTOK
613A:B1 87     188          LDA  ($87),Y
613C:09 80     189          ORA  £$80
613E:20 ED FD  190          JSR  PRNT
6141:C8        191          INY
6142:4C 34 61  192          JMP  LOOP3
6145:B1 87     193 ENDTOK   LDA  ($87),Y
6147:20 ED FD  194          JSR  PRNT
614A:E6 24     195          INC  HTAB
614C:A9 D0     196          LDA  £$D0
614E:85 87     197          STA  COML
6150:85 88     198          STA  COMH
6152:AC 70 61  199          LDY  SPRE
6155:B1 85     200          LDA  ($85),Y
6157:C9 81     201          CMP  £$81
6159:D0 06     202          BNE  NFOR
615B:EE 6C 61  203          INC  INCR
615E:EE 6C 61  204          INC  INCR
6161:          205 NFOR     EQU  *
6161:C9 82     206          CMP  £$82
6163:D0 06     207          BNE  NNXT
6165:CE 6C 61  208          DEC  INCR
6168:CE 6C 61  209          DEC  INCR
616B:          210 NNXT     EQU  *
616B:60        211 END      RTS
616C:          212 *
616C:00        213 INCR     DFB  $00
616D:00        214 NXAL     DFB  $00
616E:00        215 NXAH     DFB  $00
616F:00        216 TOCT     DFB  $00
6170:00        217 SPRE     DFB  $00


*** SUCCESSFUL ASSEMBLY: NO ERRORS
```

Deeper into *tryit*

Part 8 of Stuart Bell's tutorial series covering the unitary approach to program development

LAST month we started our development of the *Apple User* File Control Unit by discussing how we can access the variables within the operating system which we need to inspect, and perhaps even change, in order to write a unit which will perform the functions of the system's Filer.

We finished by looking at a program called *tryit*, which demonstrates the techniques involved, and we shall now turn to that program in more detail.

The procedure *getsyscom* finds the start of the variable area in the operating system, as we saw last month. The first variable is called the *system communications area*; it is of type *syscomrec*, and *getsycom* sets *syscom* to point to it. That is, *syscom* will contain the address of the data area in which we are interested.

Now look at Figure I. This shows the layout of variables within the global declarations, with their offset from the start. It shows, for example, that the date is located 132 bytes beyond syscom. Hence, to find the date, we simply peek at the locations 132 bytes beyond the address held in syscom.

Integral to the success of our technique is the use of variant records to effectively peek at memory locations within the Apple. (Note that while I have in the past discouraged such things, here we are not accessing fixed locations, but rather following pointers within the system to find the required data.)

We shall only run into problems if Apple change either the layout of the global declarations of the operating system – and hence the offsets which we use – or the location of the p-machine register MSSTAT which is currently 82).

Since the program has been tested on Apple Pascal 1.1, 1.2 and 1.3, compatibility should be no problem. Note, however, that SYSCOM does change between versions; hence we do have to follow the MSSTAT chain in order to find the System's global data area.

To remind ourselves how variant records work, look at the declarations in *tryit* for the type *datatype*, the variable *thedate*, and also the code at the start of the main program. *Thedate* may either be an integer, or a pointer to a record holding the date. The statement:

```
thedate.addr:=syscom+thedateoffset;
```

treats it as an integer, and sets that integer to the address of the date, as we have just discussed.

Then, the statement:

```
with thedate.ptr^ do
  writeln('Today is: ',day,'/',month,'/
',year);
```

treats it as a pointer to the record, and permits access to the record at that location in a structured manner. This is a much cleaner approach than doing an explicit peek of the bytes, then having to work out what the values mean in terms of the actual date. A similar approach is used to access the other variables within the operating system's data area.

Thus, *userinfo.addr* is set to point to the data area of type *inforec*, which holds whether there exists a system text file and code file (*gotsym* and *gotcode*), and the names of these files.

Similarly, *unitable.addr* points to the unit table, an array which holds details of every unit (in the sense of input/output device, not program unit) which is online.

Because discs may have been changed since the system last checked each disc-drive, the program *tryit* loops through the volume numbers, and for each volume which is a blocked volume (that is, a disc drive or pseudo-disc drive), it tries to open a file called *dummy*. This will make the system update the unit table. If a file can be opened, it is then closed with "purge" to delete it. This is a safer technique than accessing the directories directly, and makes sure that the system is kept up to date.

We can also find the largest space on each blocked volume. When the system opens a new file on a volume, it will (unless told otherwise), use the largest space available. Thus, if we can find the size of the new file *dummy*, that is the size of the largest space.

Now whenever the system opens a file it also creates a File Information Block (FIB), which is used to store housekeeping information about the file, together with the window variable into which each record of that file is read from disc. In it are stored such things as the size of each record, and – the ones we want – the next block to be read and the number of the last block in the file.

When the statement:

```
rewrite(f,fn);
```

is executed, an FIB is set up for the file. The variable *fibarea* is simply an array of integers, into which we shall copy the FIB, so that we can examine the data held there.

The effect of:

```
moveleft(f,fibarea,36);
```

is to copy the contents of f – the FIB of the file – into our area *fibarea*, without worrying about the structure of the data being moved. The manuals (1.1 Language Manual p.52) says that the variables being moved must *not* be a file type. This must be a restriction intended to stop naughty programming practices, rather than one actually prohibited by the compiler, for the operation is allowed, and works – indeed, I have seen this technique used within the source code of part of an early version of the p-System. The 1.3 manual makes a similar comment (p. III-95), but the rule is not enforced.

Once *fibarea* has been made to store the

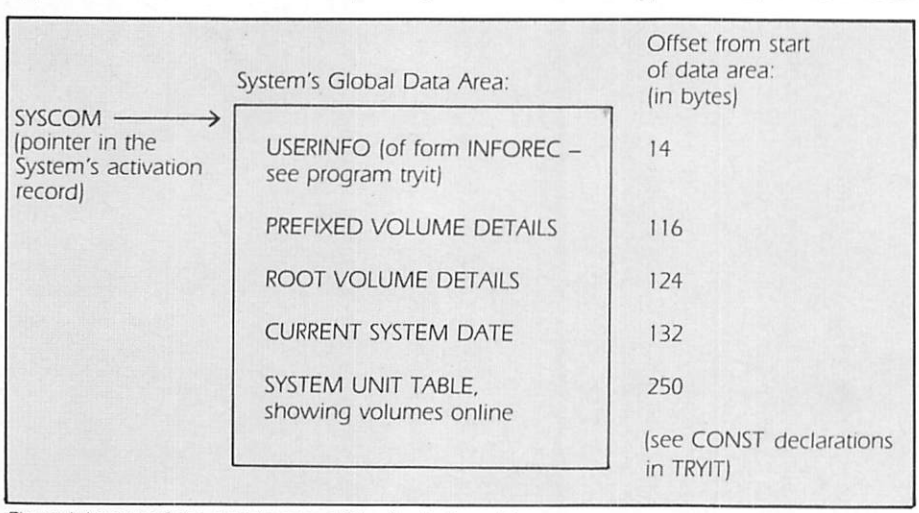


Figure I: Layout of the globals area of the Apple Pascal System

FIB, we can find the length of the file by subtracting the next block to be read (that is, the first one, if no reading has yet been done) from the number of the last block. This is achieved by the statement:

```
largespace[i]:=(fibarea[17]-
fibarea[16]);
```

Thus, we can now build up a picture of much of the internal data area of the operating system, without resorting to manually accessing the discs with *unitread* and (more dangerously) *uniwrite*. Program *tryit* will display the date, a list of volumes on-line, and available space.

If you haven't already done so, type in *tryit*, and run it. Much of its code will be reused in the File Control Unit itself, to which we now turn.

Turn to pages 68-69 of the 1.1 OpSys Manual or pages II-24,25 of the 1.3 manual. Here is the task to be attempted. The Interface part of the *Apple User* File Control Unit is given in Listing I. The constant and type declarations are here in the Interface – and hence public – so that programs using the Unit can also make use of them. The two files it uses have to be declared here, because private files are not allowed.

Then follows a list of the Filer operations being provided, each of which is followed by the functions which either perform the task, or a related function that is more relevant to a Unit.

For example, while K(runch performs the exact Filer function, the functions *isonline* and *volnameof* are rather more useful to calling programs than a simple volume lister would be. (Should you need that function, it was of course provided in last month's demonstration program). The comments against each function should be pretty clear: Next month we'll start looking at how they are implemented.

Any one interested in obtaining the source listings of this series of articles should send a disc and SAE to Max Parrott, Apple User, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.

```
(*$S+ *)                (* invoke swapping to compile a Unit *)
Unit FileControl;

Interface

CONST MAXUNIT = 12;
       MAXDIR = 77;
      VIDLENG = 7;
      TIDLENG = 15;

TYPE  DATEREC = PACKED RECORD
                  MONTH: 0..12;
                  DAY: 0..31;
                  YEAR: 0..100
      END;

      UNITNUM = 0..MAXUNIT;
          VID = STRING[VIDLENG];
     DIRRANGE = 0..MAXDIR;
          TID = STRING[TIDLENG];

VAR          f:file;
          destf:file;

(* Now follow the declarations for the functions provided by the Unit.
   They are grouped according to the Filer operations which they either
   simulate or to which they provide related functions. The order is that
   on pages II-256,257 of the Apple Pascal 1.3 manual. *)
```

```
(* Volumes                *)
     function isonline(volno:integer):boolean;
     (* returns whether or not a particular unit is on-line *)

     function volnameof(volno:integer;var errno:integer;
                                                var volname:vid):boolean;
     (* returns name of volume associated with specified unit number *)


(* List-directory         *)
(* Extended-directory-list *)
     function listdir(volno:integer; var errno:integer):boolean;
     (* lists the directory on the screen: filenames only, 4 columns *)

     function fileonvol(volname:vid; fname:tid; var errno:integer):boolean;
     (* returns whether or not specified file is to be found on that volume *)


(* Krunch                 *)
     function krunch(volno:integer; volname:vid; var errno:integer):boolean;
     (* compresses file on specified volume.
        As a safety measure, both volno and volname must be correct *)

(* Zero                   *)
     function zero(volno:integer; oldvolname:vid; newvolname:vid;
                                              var errno:integer):boolean;
     (* zeroes specified volume. As a safety measure, both volno and oldvolname
        (if able to be read) must be correct *)

(* Prefix                 *)
     function prefix(volname:vid; var errno:integer):boolean;
     (* sets prefix volume to the name in volname *)

(* Transfer               *)
     function transfer(fromvol,tovol:vid; fname:tid; var errno:integer):boolean;
     (* transfers file of name 'fname' from volume 'fromvol' to volume 'tovol' *)

(* Make                   *)
     function spaceon(volname:vid; var errno:integer;var space:integer):boolean;
     (* returns largest space on specified volume *)

     function make(volname: vid; fname:tid; size:integer;
                                        var errno:integer):boolean;
     (* creates file with specified details on volume stated*)

(* Change                 *)
     function change(volname:vid; oldfname,newfname:tid;
                               var errno:integer):boolean;
     (* changes file name as specified *)

(* Remove                 *)
     function remove(volname:vid; fname:tid; var errno:integer):boolean;
     (* removes specified file from specified volume *)

(* Get                    *)
     function getfile(volname:vid; fname:tid; var errno:integer):boolean;
     (* gets specified file as workfile - ignores filetype suffixes *)

(* Save                   *)
     function savefile(useoldname:boolean;fname:tid;var errno:integer):boolean;
     (* saves file: if useoldname is true, uses stored symvid and symtid,
        otherwise uses newly specified names. Must be on root volume  *)

(* New                    *)
     function newfile(var errno:integer):boolean;
     (* clears workfile, deleting SYSTEM.WRK= if they exist *)

(* What                   *)
     function whatfile(var iswrkfile:boolean; var volname:vid; var fname:tid;
                                                var errno:integer):boolean;
     (* returns name of workfile (if any ) *)

(* Bad Blocks             *)
     function badblocks(volno:integer; startblock,stopblock:integer;
                    var errno:integer; var firsterrblock:integer):boolean;
     (* does a bad block check on the specified volume, returning the
        number of the first bad block found. Errno set to 0 if none *)

(* Examine:      This is too dangerous to use without user supervision! *)

(* Date                   *)
     function olddate(var errno:integer; var date:daterec):boolean;
     (* returns the current date as held in the system's global variables *)

     function newdate(date:daterec; setdisc:boolean; var errno:integer):boolean;
     (* sets date held in memory to newdate, changing boot disc's record
        of the date if 'setdisc' is true.                              *)

(* Quit:    nothing to do!                                             *)
```

National Apple User Groups:

Apple 2000. The Apple User Group, P.O. Box 177, St. Albans, Hertfordshire AL2 2EG. Tel: 0727 73990.

MacTel. Bulletin Board for the European Macintosh Community, 15 Elm Tree Avenue, West Bridgford, Nottingham NG2 7JU. Tel: (voice) 0602 810237.

The Macintosh User Group UK. The UK's largest Macintosh User Group. The professional organisation with local groups. 55 Linkside Avenue, Oxford OX2 8JE. Tel: 0865 58027.

Local User Groups:

MacCam Macintosh User Group (Cambridge). Patrick Winterson. Tel: 022026 2436.

Suffolk & Cambs Gateway Computer Club. Robert Hall. Tel: 0638 717723 (Any time).

Berks & Hants Apple User Group. Mike Hollyfield. Tel: 0734 780301 (Evenings & Weekends).

Midapple. Tom Wright. Tel: 0527 71913.

Herts & Beds Apple and Macintosh Computer Group. Norah Arnold. Tel: 0582 573918.

Cambridge Apple User Group. Ian Archibald. Tel: 0223 311157.

Midland Mac. Ivan Knezovich. Tel: 0299 403418.

London Apple Computer Club. Chris Williams. Tel: 01-882 0333.

Bristol Apple Users & Dabblers. Michael Farmer. Tel: 0272 230000 ext. 2585 (Day).

Croydon Apple User Group. Graham Attwood. Tel: 01-850 5622 (Evenings & Weekends).

North-West Apple Users Group. Max Parrott. Tel: 061-236 3311 ext. 2055 (Day) 061-432 3487 (Evenings).

Apple Crackers Bulletin Board. Mike Jones. Tel: 0268 779244 (Evenings).

London Macintosh Users' Group. Maureen de Saxe. Tel: 01-458 4890.

North West Apple Computer Club. Jim Roscoe. Tel: 0925 38101 (Evenings).

Essex Apple User Group. Patrick Bermingham. Tel: 0245 261636.

Kent Apple Computer User Group. Richard Daniels. Tel: 0303 60515 (Day) 0303 58349 (Evenings).

Liverpool Apple User Group. Irene Flaxman. Tel: 051-928 9097.

Computers Unlimited Macintosh User Group. James Sanson. Tel: 01-349 2395 (Day).

Bentwaters Apple User Group. John Thomas. Tel: 0394 270240.

You're never alone with an Apple user group.

There are various Apple™ user groups dotted around the country.

And if you're not a member, you're really not getting the most out of your Apple.

For example, many of us publish regular magazines, full of news, views and reviews of new software. Special deals on blank disks and shareware abound. Hot tips on how to solve problems litter the pages. Some of us even have a phone 'Hotline' service.

We also hold meetings where you can try out various bits of hardware and software, listen to speakers on assorted Apple subjects, or just chat with other Apple fanatics!

We maintain links with Apple Dealers, software manufacturers and Apple Computer itself, but we're not beholden to any of them. And we attend all the Apple events and new product launches, to keep you up to date.

It's incredibly easy to join – just phone up your local group, and we'll give you all the details.

You've already got the only qualification you need... you've got an Apple computer!

The chameleon chip

Stuart Bell takes an in-depth look at the 65C816

WHEN the Apple IIgs was first announced (or, perhaps, when it was first leaked), there must have been many who reacted to the news of its microprocessor by saying, "The 65... what?" Since then, it has been described as a "super 6502", or a "6502 with an extended address range". In fact, the 65C816 and its sister device the 65C802 are very complex pieces of silicon, designed to be switchable between pretending to be good old 6502, and being a pretty powerful 16 bit microprocessor.

We must remember, however, that these two descriptions are mutually exclusive. At any one time it can be one or the other. In that sense, it is certainly not a 16 bit 6502. In consequence, when its being a 6502, that's really all it is, and running at say 2.8MHz – as it does in the Apple IIgs – it will perhaps be a little slower than a REAL 6502 in a accelerator running at 3.6MHz.

Before looking in any more detail at this chameleon chip, let's back track a few years to see where the 65C816 comes in the family tree of microprocessors. In about 1974, there were but two widely-used 8 bit microprocessors, the Intel 8080 and the Motorola 6800. Both were used in early hobbyist computers, for example in the Atari 8080 and the SWTPC 6800 machines.

Question of cost

From the 6800 a firm called MOS Technology developed the 6502, a device with many similarities to the 6800, but with arguably more power. It was this device that Messrs Wozniak and Jobs used in the famous Apple I computer (largely, it is said, on the grounds of cost). From then on the Apple II range of II, II+, III, IIe, and IIc were all 6502-based machines – along with those from Commodore and a host of other companies, the majority of which are no longer with us.

The 8080 was enhanced by Zilog into the Z80, and followed by the whole Intel range of 8088, 8086, 80186, 80286, 80386, which are now the processors at the heart of the PC/MS dos market. Motorola's 6800 had not been much of a commercial success in micros, so their engineers could make a clean start with the 68000 family, now seen in the Macintosh, the Atari St range and the Commodore Amiga.

This historical background is relevent because it sets the context for the the arrival of the Apple IIgs with its 65C816 processor. In the period from the 6502's announcement in 1975 until the 65C816, the 65xx line of processors had undergone only minor development. One or two 6502 manufacturers had hinted at the development of a more powerful member of the 6500 family, but none made it to the production stage.

However, by last year, the micro market was dominated by one very powerful force, and one rather less powerful one. In other words the PC-clone vs. the Apple Macintosh battle was well under way. There is no volume micro that does not have as its heart either a member of the 808x/80x86 series, or else a 68000 family member. Except, of course, for the Apple II series.

Faced with declining sales as Apple IIs were overtaken by the potential power, memory size, and software range of the MS Dos world, Apple turned to the product of a relatively unknown company, the Western Design Centre, with a virtually unknown device, the 65C816, to reassert the position of the Apple II range in the market place. (Note that the Western Design Centre should not be confused with Western Digital, another semi-conductor manufacturer most famous for its sophisiticated disc-interface ICs.)

The key to understanding this choice is clear; the ability of the 65C816 to simulate (that is behave like) a 6502, and thus allow it to run the vast range of Apple II software which has been written. Furthermore, the ability of the 65C816 to go beyond that into that into the true 16 bit world make possible applications for which the limited power and, more particularly, the limited addressing range, of the 6502 was inadequate. The 65C816 will, it is hoped, both extend the life of the Apple II range, and permit the partial harmonisation of the Apple II and Macintosh ranges by the provision of common peripherals and a common user interface.

So much for the theory. Let us now examine this device on which so much of Apple's future depends. It would be naive to suggest that chip architecture alone will determine the success of the IIgs but it will affect the amount and quality of software written for it – and that will determine whether the IIgs takes off as the "odd one out" in this 808x/68000 dominated world.

Three factors describe a microprocessor, and the power which it offers to the programmer: Architecture, Instruction set, and Addressing modes.

By architecture, we mean the layout of the registers within the device, and the way in which the processor is connected to other parts of the system, such as memory or peripheral devices. In general, the more registers a processor has, the larger they are. And the more flexibly they can be used, then the more powerful a processor is said to be. The registers of the 6502 are shown in Figure I. Immediately, the weaknesses of the design for a modern micro can be seen: The accumulator – that register in which all arithmetic and logical operations are performed – is only 8 bits long. Thus, only numbers in the range 0-255 can be operated upon at any one time, and operations on larger numbers numbers are a laborious process.

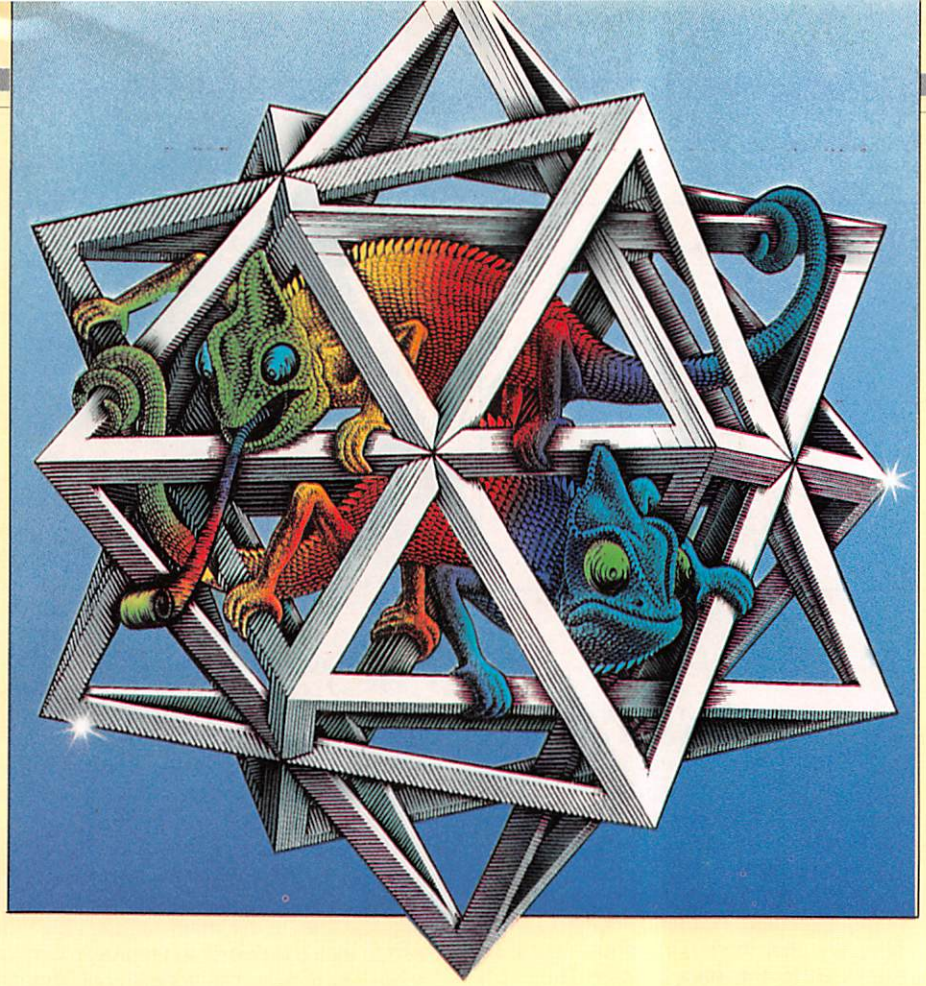
Operations limited

While there are two index registers (X and Y), an improvement over the 6800, they are again only 8 bits long. Thus, operations which use them, especially in the accessing of tables of information, are limited to data areas up to 256 bytes long, unless extra operations are performed to jump to the next 256 bytes of data.

The stack pointer is also limited to 8 bits. This limits the stack to 256 bytes, which in the 6502 are located in the area from 100

Accumulator (8 bits)
X Register (8 bits)
Y Register (8 bits)
Stack pointer (8 bits)
Program counter (16 bits)
Status register (8 bits)
S V B D I Z C

Figure I: 6502 register layout



(hexadecimal) to 1FFh. This makes the use of the stack to store a large number of values difficult, and makes very messy the implementation of multi-tasking environments in which several active programs may each wish to have their own stack. This is usually achieved by changing the stack pointer for each task.

The program counter which holds the address of the next instruction to be obeyed is 16 bits long. The implication of this is that the addressing range of the 6502 is only 65536 bytes. In other words the largest amount of memory that can be directly accessed by the 6502 is 65536 bytes. Of course, the expanded IIe and the IIc, not to mention IIs with various ramcards, can access more; but various, inelegant, techniques such as bank-switching have to be used to do this. No 6502 can access more than 65536 bytes cleanly at any one time.

Finally, the status register holds a number of flags – one bit locations which indicate the state of various things within the system, for example if the accumulator is holding a zero. We shall see later how the 65C816 has expanded the status register to offer more flexiblity.

Powerful facilities

With such a damning description of the 6502, one might wonder how machines based on it ever succeeded. The answer, of course, is that by the standards of the 1970s the 6502 was a pretty neat device, with some very powerful facilities. An example was its ability to treat the first 256 bytes of memory (the so-called zero-page) as if it consisted of 128 16-bit registers for the purpose of pointing to other memory locations. However, the 6502 – whatever clever tricks are employed – could never compete with something like the 68000 with its sixteen 32 bit registers.

Thus the 65C816 enlarges the register layout of the 6502, primarily to increase the size of existing registers. To describe the change is not simply a case of drawing another diagram, because the effective register layout of the 65C816 varies according to in which one of three modes it is currently running. In Emulation Mode (that is emulating a 6502), the available registers are those shown in Figure II. Note that it is not quite the same as that of the 6502.

Also, as we shall see later, since all instructions and addressing modes of the 65C816 are available, a 65C816 nominally emulating a 6502 is in fact rather more powerful than a true 6502.

The differences are obvious: Firstly, there are two accumulators, called A and B. The former is the equivalent of the 6502's accumulator. While in emulation mode, B cannot be acted on directly: There are some instructions in this mode which allow access to it.

The index registers are shown as 16 bit registers (as they really are when not in emulation mode), but with the top byte set to 00. The effect is that they are 6502-like 8-bit registers. Similarly, the stack pointer is shown as being a 16 bit register, with the top byte set to 01 so that the effect is just like that of the original 6502 SP.

The program counter is, as one would expect, 16 bits long, just like the 6502's, and gives access to 64k of memory. The status register is slightly different; all that we need note is that the emulation bit is set to 1 when the 65C816 is pretending to be a 6502. The emulation bit is not directly accessible. it is shown under the carry bit because one instruction allows us to swap those two bits.

You will have noticed the two extra 8 bit registers in Figure II: A data bank register and a program bank register. These are the registers which are the key to the power of the 65C816 when not in emulation mode, because they extend the range of memory which the device can handle from "2 to the power 16" to "2 to the power 24" – about 16 million bytes. While they can be operated upon in emulation mode, it is a rather futile exercise as they are ignored for the purposes of accessing data, so that we are limited to 64k of memory.

Finally, there is the direct register. We have noted the power of the 6502's zero page addressing modes: what the direct register does in emulation mode is to allow access to any part of the 64k of memory in the same way that the 6502 can access the first 256 bytes. What happens is that the address in the DR is added to the offset in the instruction.

For example, if the instruction specified location 40h, and the DR contained 1000h, the operation would be performed on location 1040h, just as if it were in zero page. Note that whether wrap round occurs – that is if when the lower byte of the DR and the address in the instruction are added together, they are allowed to increment the top byte of the effective address – depends on the particular instruction. The direct register is potentially very powerful. For example it would effectively allow each of a number of multi-tasking programs to ▷

Accumulator A (8 bits)
Accumulator B (8 bits)
X register (8 bits)
Data bank register
Y register (8 bits)
Direct register (16 bits)
Stack pointer (8 bits)
Program bank register
Program counter (16 bits)
Status register (8 bits)
S V 1 B D I Z C
E
(Emulation bit – 1)
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1

Figure II: 65C816 register layout emulation mode:

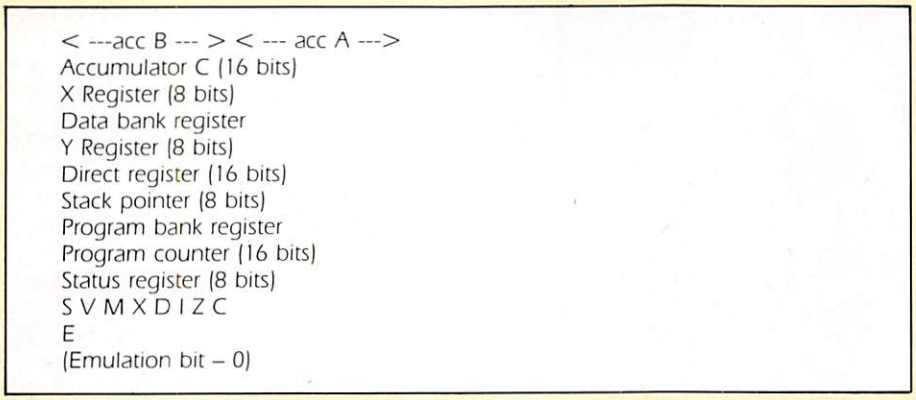


Figure III: 65C816 register layout in native mode

◁have their own zero pages.

When not in emulation mode, the 65C816 is termed as being in native mode. In this mode, the device becomes a 16 bit microprocessor, with the ability to generate 24 bit addresses. However, within native mode it may operate with either 16 bit or 8 bit registers. The M bit in the status register makes the processor treat the accumulator as an 8 bit register. The X bit (index register select) has a similar effect on the index registers. In emulation mode, both M and X bits are forced to 1, as the 6502 only has 8 bit registers. We shall effectively ignore this half-way-house between an 8 bit and a 16 bit processor, and consider the 65C816 only when running in proper native mode. The register layout is shown in Figure III. It is very similar to that of the 65C816's emulation mode, but the differences are important:

The status register has grown slightly; we shall not consider it in detail now, but wait until we look at some addressing modes which use it. Note that E is now set to 0 as it is not emulating a 6502. The X bit replaces the 6502 BRK bit. We have noted its use, and that of the M bit as dertermining effective register length.

Not restricted

More significantly, X, Y and S are now all genuine 16 bit registers. This means that they are no longer restricted to accessing 256-byte sized pages of information. Thus, the stack can be anywhere in a 64k area, and is not limited to locations 100h to 1FFh, as is the 6502's

Associated with the X and Y registers are two data bank registers. In the precise technical way that memory bank is used for the 65C816, it is a new term to 6502 users. In the same way that the 64k address space of the 6502 was considered to be divided into 256 PAGES of 256 bytes each, so the 16Mb space accessible by the 65C816 is considered to be divided into 256 data BANKS, each of 64k. Thus, in these terms, the 6502 could access one bank of memory.

As you may have guessed , the data bank register defines on which bank of memory an operation with an addressing mode which refers to either the X or Y registers will operate. Thus, if the DB is set to 7, an operation using the X register will access memory bank 7, located just within the first half-megabyte of memory.

Similarly, the program bank register defines in which page of memory the code being executed is being located. Note, however, that there is no equivalent 8 bit extension to the stack pointer. Thus, it is limited to 16 bits, and the stack must always be located within the first 16k of memory – that is in the first page.

You may have wondered why the index registers and the program counter were not shown as 24 bit registers, since together with their extensions they are effectively 24 bits long, and can effectively access 16Mb of memory. The answer is very simple, and one that definitely limits the power of the 65C816.

The two bytes of each of the registers are effectively linked together, so that if the lower byte contains FFh, and is incremented, then as well as it becoming 00, the higher byte will be incremented correctly. For example if X contains 65FFh, and is incremented, it then stores 6600h.

However, the extension registers are not connected in the same way. If the data bank register contains 77h, and the X register contains FFFFh, when the X register is incremented, it will become 0000, but the DB register will remain at 77h. In short, the extension registers are not connected to their 16 bit associates in the same way that the two halves of those registers are linked.

The effect of this for the use of the X and Y registers is clear. Data can only be accessed uniformly within units of 64k, or one page. Hence it will be difficult to implement a data structure (for example a large array) larger than this. Also, data areas that cross a bank boundary will be very difficult to manipulate. This is analagous to the problem of crossing 256byte page boundaries with the 6502 and of crossing 64k boundaries with the 8086 found in MS Dos machines.

The effect of the fact that the program bank register is not linked to the program counter is perhaps more significant. It means that no program, or to be more general no piece of code, can be longer than 64k. Of much greater importance is the fact that no piece of code can be loaded into memory so as to straddle a page boundary. If a program was loaded like that, when the processor reached the page boundary, instead of executing the first instruction of the present page, it would go back to the first instruction in the next page – that is it would go back 64k in memory!

This makes the efficient management of memory, and the writing of large programs using large areas of data (such as a very large spreadsheet), quite difficult. In comprison, the 68000 allows access to its 16Mb through true 24 bit registers (internally they are actually 32-bits), and hence without any such limitations of page size.

On the other hand, it must be admitted that the 8088/8086 has a similar limitation (probably because it too was derived from an 8 bit processor), and there is no shortage of powerful software for the machines that use that device.

The issue is one of ease and cleanliness, rather than one of what such a segmented or paged architecture makes impossible. Because the 16 bit main registers are not tied to their 8 bit extension registers, I'm not really sure that we can honestly talk about a 24 bit address bus. Certainly it's 24 bits long, but the 24 bits are not as closely linked as such a description might suggest. A linear address space is much to be preferred to this segmented approach.

Emulation mode

Finally, we note perhaps the most important change for arithmetic and logical operations. The accumulator has grown to 16 bits. It is composed of the two 8 bit accumulators available in emulation mode. Thus, all operations such as ADC or AND can be performed on two bytes at a time. All logical operations on any data larger than 1 byte will be significantly faster, and require less code. Thus, the 65C816 is quite rightly described as a true 16 bit processor.

Do not forget, however, that this only applies in native mode. With the accumulator in emulation mode the 65C816 remains an 8-bit device, and programs written for the 6502 are simply unable to take advantage of the improved architecture of the 65C816.

To take an analogy from the motoring world, I could replace my Mini by a Lamborghini Countach to collect my shopping from the supermarket, but such use wouldn't really use its capabilities to the full: an emulating 65C816 is just a plain 6502 at heart.

● *That completes our review of the register layout of the 65C816. Next month we'll see how it is interfaced to the outside world, including one version, the 65C802, which brings some of the power of the 65C816 in a device that is a plug-in replacement for the familiar 6502.* □

Application Form

Name
Position
Company
Address
Postcode Daytime tel:

Commencement of Service
Please indicate month of commencement ☐☐ 19☐☐
Allow 7 days for validation of mailbox

Date of first payment to be on 15th of month following commencement. Please complete billing authorisation form A, B or C.

* *Telecom Gold is a trademark of British Telecommunications plc.*

I/We hereby apply to join MicroLink

(✓)☐ I enclose my cheque for £5 payable to Database Publications as registration fee to MicroLink.

(✓)☐ I also wish to use Telex. I authorise you to charge an additional £10 to my initial bill for validation.

☐ I confirm that I am over 18 years of age.

☐ I confirm that I accept the terms and conditions for the time being in force, copies of which are available on request

I intend to use the following computer model _____

Signature _____

Date _____

A. Direct Debiting Instruction *(Enter full postal address of UK Bank Branch) to:*

I/We authorise you until further notice in writing to charge to my/our account with you on or immediately after 15th day of each month unspecified amounts which may be deited thereto at the instance of Database Publications Ltd – MicroLink by Direct Debit. Bills are issued 10 days before debit is processed.

Name of Account to be debited
Account Number Sort Code

B. Please debit my/our Access/Mastercard/Visa/ *American Express No.
**Overseas subscribers only*

I/We authorise you until further notice in writing to charge to my/our account with you on or immediately after 15th day of each month unspecified amounts which may be debited thereto at the instance of Database Publications Ltd – MicroLink. Bills are issued 10 days before charge is applied to your account.

Signature _____ Date

C. Please invoice the company/authority.

(✓)☐ If you select this option, which is ONLY AVAILABLE to government establishments and Public Limited Companies, we will require an OFFICIAL ORDER to accept unspecified amounts.

Send to: **MicroLink, Database Publications, Europa House, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.**

FOR OFFICE USE ONLY:
Mailbox assigned _____
Start date _____
Password _____

AU6

Personalised code systems

R.H. Brown's encoding series continues with the steps to follow in designing a coding system

Earlier parts of this series offered a program to input characters to base 36 (0 to 9 and A to Z), processing characters four at a time, first converting to base 10 and finally to any base between 11 and 35.

Extra characters were added to make it difficult to decide which base or decoding key was used. Examples of coding APPLE USER are given in Figure I.

Many modifications are possible, including the following:

● Process characters in groups of one, two or three

Considering a group of characters to base 36, the largest value is:

ZZZZ = 36^4-1 = 1 679 615 to base 10.

To be able to encode a base B it is necessary for the largest possible value to base B to be greater than, or exactly equal to 36^4-1.

Suppose this requires n characters: Then the maximum value for n characters to base B = B^n-1.

$B^n-1>=36^4-1$

$B^n>=36^4$

n log B>= 4 log 36

or

n >=4 log(36)/log (B)

For base 6, n=4 log(36)/log(6)=8 (exactly): For other bases, the calculated value of n will not be an integer. INT(4 log(36)/log(B)) will therefore round down for every value of B except 6, so that:

n = INT(4 log(36)/log(B))+1

which is equivalent to:

MC% = 4*log(36)/log(BB%)+1

To change the group size to three, it is therefore only necessary to change line 530.

```
530 MC% = 3*log(36)/log(BB%)+1
```

A list of corresponding group sizes for two,

three or four characters to base 36 is shown in Figure II.

● Use a different group size per block for alternate blocks

A decoding key, 341817, could represent alternate blocks of three characters coded to base 18, followed by four characters coded to base 17 with the sequence repeated for the complete line. Using 341817 for APPLE USER:

Group of three characters to base 18
APP

Group of four characters to base 17
LE U(that is, LEZU)

Group of three characters to base 18
SER

Figure III is used to assist in conversion to base 10.

APP_{36}
$= A \times 36^2 + P \times 36 + P$
$= 10 \times 36^2 + 25 \times 36 + 25$
$= 13885_{10}$ (five characters)

13885_{10}
$= 2 \times 18^3 + 6 \times 18^2 + 15 \times 18 + 7$

$= 2\ 6\ F\ 7$ (four characters)

$LEZU_{36}$
$= L \times 36^3 + E \times 36^2 + Z \times 36 + U$
$= 21 \times 36^3 + 14 \times 36^2 + 35 \times 36 + 30$
$= 0999210_{10}$ (seven characters)

0999210_{10}
$= 0 \times 17^5 + 11 \times 17^4 + 16 \times 17^3 + 6 \times 17^2 + 8 \times 17 + 1$

$= 0\ B\ G\ 6\ 8\ 1$ (six characters)

SER_{36}
$= S \times 36^2 + E \times 36 + R$
$= 28 \times 36^2 + 14 \times 36 + 27$
$= 36819_{10}$ (five characters)

36819_{10}
$= 6 \times 18^3 + 5 \times 18^2 + 11 \times 18 + 9$
$= 6\ 5\ B\ 9$ (four characters)

The coded message is therefore: 26F70BG68165B9

● Put some spaces back in the message

All previous codes converted spaces to the character Z before encoding. However, in the final code, spaces could be inserted as dummy characters to give the appearance of separate words.

The message in Figure IV has a decoding key of 45241327 with the following rules:

45 24 1327

leading chrs. group sizes two bases used

● Dummy characters are inserted according to the higher of the two bases used. In this case, the bases are 13 and 27. Dummy characters are from R to Z, since base 27 uses characters up to and including Q.

● Odd numbered lines: Four leading characters, then alternate groups of two characters coded to base 13 followed by four characters coded to base 27.

● Even numbered lines: Five leading characters, then alternate groups of two characters coded to base 27 followed by four characters coded to base 13.

● Spaces are inserted at random on each line.

The present program could be adapted to decode Figure IV, but it can be done – in time – by pencil and paper. In any case, a little originality will produce a personalised code system that will be very difficult to break.

Decoding Key	Coded Message
418	LIE1WWL4DORCF36TSMSQBJHJ7A6VZAK3HH
819	DWAK9FWW3VWFGZTMTDKAL5701SF505Y9MTGI
921	6ZVKEYOLJ2YBSKXAI3CBATNZG3XBPY98KV
628	C3FIZM0MLZGPZ1W30BG13BBRT
534	70Z390C0EYD0HR800HIEJ

Figure I: Examples of coding APPLE USER

4 Characters		3 Characters		2 Characters	
base	MC%	base	MC%	base	MC%
10	7	10 - 14	5	10	4
11 - 17	6	15 - 35	4	11 - 35	3
18 - 35	5				

Figure II: Corresponding group sizes for encoding

Base 36	Base 10 Equivalent
0	0
9	9
A	10
F	15
K	20
P	25
U	30
Z	35

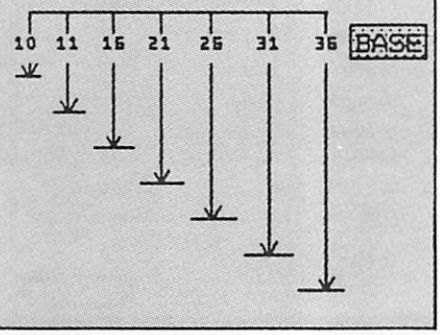


Figure III: Valid characters for bases 10 to 36

A9Y D31S71 3 1X9 Z04 CUC
4 6XC7 0ZYHN 0TK 92S A1 147
Y0 SD 564Y 2SZ 3QLA4 BT8
548S 1 Y0J 01 T449 XBY 40JQ

Figure IV: Decoding key 45241327 – for the solution see the Feedback pages

□

Integration that really Works

John Richardson reviews a five-in-one package that makes financial sense

MICROSOFT'S Works is an integrated package containing five of the most common types of office software: word processing, database, spreadsheet, charting and communications. The result is a great deal more successful than the "Jack of all trades – master of none" that you might expect.

There are two main reasons why you might be attracted by the idea of an integrated package: the general convenience and the financial savings.

Imagine that you are responsible for producing publicity material about a company. The task might involve word processing a document and merging the recipients' names and addresses from a database. In composing the letter you may need to refer to a spreadsheet covering past financial performance and copy figures from it. You might even have to dial up a remote database to check some background material.

With an integrated package like Works you can do the whole job from within one application without using the Switcher, let alone having to copy files or swap discs.

Unless you are a professional it is unlikely that you will use all the facilities found within a specialist piece of software. A chartered accountant might use all the tricks that Excel can perform but, ordinary folk like you and I probably wouldn't. However, Works' spreadsheet will let you do most of the things you could do with Excel.

Apply this argument to the other components in the package and the financial sense begins to become clear. Works costs no more than a single specialist package yet it can do most of the work of four. The crucial word is most: It is likely that many users will use one of the parts more than the rest so it's vital that the important component for you does all that you need.

First let's take a look at the separate components and then see how well they fit together as a whole.

Word processing

The Works word processor is a good compromise between the ease of use, but limitations, of MacWrite and the seemingly endless capabilities of Microsoft's own new Word-3. First impressions suggest that Works offers much the same as MacWrite – but first impressions can be deceptive, and after only a short while in use Works' extra capabilities become apparent.

MacWrite limits you to a document 7" wide but Works permits horizontal scrolling to a full 22", which allows you to exploit the ability to print documents sideways or use wide printers.

Another major difference is the way in which formatting is handled. In Works, justification, margins, tabs and line spacing are all defined at the paragraph level while in MacWrite these tend to operate at the document level with ruler insertion to bring about local changes. The Works approach, once you have got used to it, is probably more convenient.

For example, if you want to change the line spacing of a particular paragraph all you need do is place the insertion point within that paragraph and pull down the Format menu. In MacWrite a ruler would need to be inserted both before and after the selected paragraph. Works also has a Copy Format command which allows a range of paragraph formats to be handled within a single document with ease.

Graphics are also handled in a more powerful way. A pull-down menu enables simple lines and boxes to be drawn via the mouse (much as in MacDraw) and these can be used in simple diagrams with text.

In addition, Works treats graphics imported from the scrapbook as "planes" and this allows them to be moved around

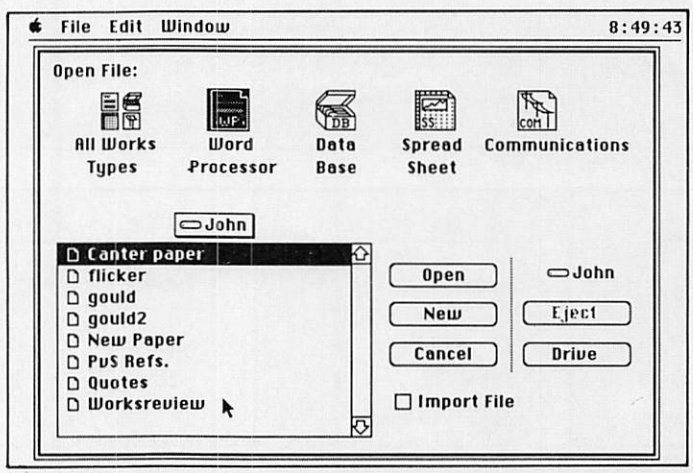


Figure I: Opening a file

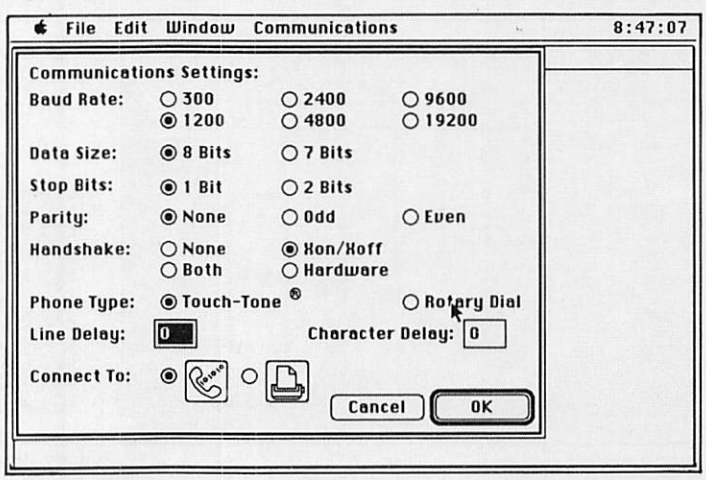


Figure II: Setting the parameters of a comms document

the page with a "grabber hand". Accompanying text can also appear on either side of graphics – a useful facility not available in MacWrite.

Two other little touches which struck me as particularly useful were the ability to Select All and to Delete: The first makes major style changes, such as reducing the point size or column width prior to printing, very easy. The second allows you to keep up with the good housekeeping without needing to close down the application – particularly useful in the context of an integrated package like Works.

One minor hiccup revealed itself when I first attempted to print a document on the Laserwriter. I had selected A4 paper and was using the default column width of 6.5".

The result? An error message and no printing, because the default page-setup specifies left and right margins of 1" each. Two inches of margin plus 6.5" of text comes out slightly wider than A4. It took only a moment to put matters to rights but the mismatch still strikes me as surprising.

Works also comes supplied with its own font, Boston, and the word processor defaults to it. While an extra font sounds like good news, Boston takes up a hefty 38k if all seven sizes are installed. My system file is usually bursting at the seams and that would mean sacrificing something else. Since Works seems to support as many fonts as MacWrite and Boston is not a LaserWriter font the benefits seem doubtful.

What does it lack? The most obvious omission is a spelling checker. Only one trial of Write Now was enough to convince me how useful a well-designed checker could be.

Spreadsheet

The Works spreadsheet stands in much the same relationship to its big brother, Microsoft's Excel, as the Works word processor does to Word-3. The important features are all there, including chart generating facilities, but if you want all the advanced facilities then you may be disappointed.

Works can handle spreadsheets with up to 9,999 rows and 255 columns, which should meet most people's requirements. This potential is supported with a powerful range of built-in functions. There are over 50 and they are drawn from the usual fields of mathematics, statistics, trigonometry and finance as well as general purpose logical functions.

Entering data into a Works spreadsheet is easy and is consistent with the database. The data is entered by simply selecting a cell and typing. The data appears in an entry bar at the top of the spreadsheet, and pressing Return or selecting the confirmation button completes the action. Cells can be selected either by pointing and clicking with the mouse or by using the cursor keys.

Data can be defined as monetary, percentages or scientific (mantissas and exponents) as well as standard numeric. Although text can only be entered as Geneva 9 point, it can be made bold or underlined so that column headings can be made to stand out from their contents. All the cells can have their contents aligned to the left, right or centre.

Once data has been entered it is easy to rearrange it in a variety of ways. Ranges of cells can be moved from one location to another without using the clipboard and a cell's contents can be repeated across a whole row or down a complete column.

The equations for calculated cell values are built up in the entry bar and this process is simplified by allowing cells to be specified simply by selecting them with the mouse and by allowing the functions to be pasted straight in.

There is a degree of control over calculated cell values since the values can be calculated automatically as the dependent data is entered or held until a command is issued. It is also possible to switch between views of the spreadsheet that show cells containing values or the formulae that determine those values.

A particularly useful feature is the ability to split the spreadsheet into "panes" – this is really stretching the windows analogy. Each pane is a vertical or horizontal subsection of the spreadsheet that can be scrolled separately. They allow parts of the spreadsheet – which otherwise would be too far apart – to be shown on-screen together.

The charting options allow for pie charts, line graphs, bar charts, superimposed bar charts and combined bar and line graphs. This list sounds impressive but producing any of them highlights an irritating feature.

A range of cells in Works must be defined in terms of its column or row numbers (such as B4:B56). It would be a lot more convenient if names could be used instead (for example "Overheads"), as with Lotus' Jazz, when constructing equations and charts.

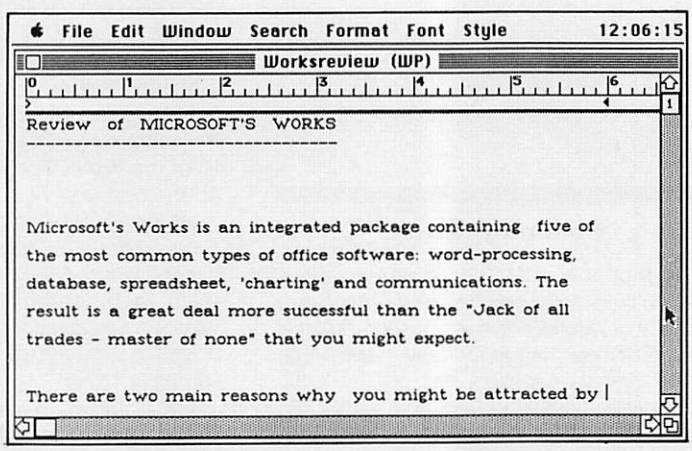
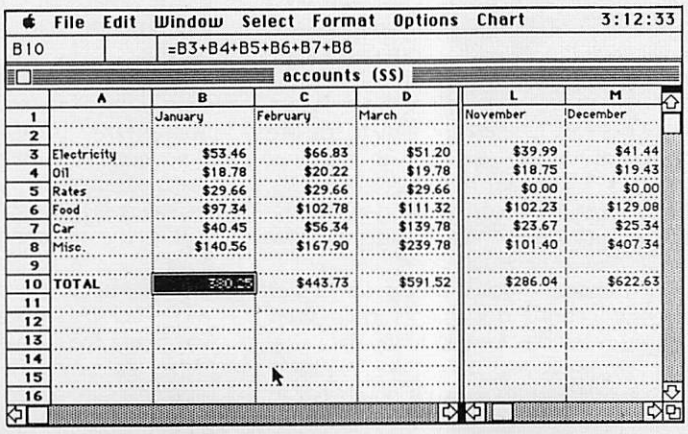


Figure III: The word processor page



	A	B	C	D	L	M
1		January	February	March	November	December
2						
3	Electricity	$53.46	$66.83	$51.20	$39.99	$41.44
4	Oil	$18.78	$20.22	$19.78	$18.75	$19.43
5	Rates	$29.66	$29.66	$29.66	$0.00	$0.00
6	Food	$97.34	$102.78	$111.32	$102.23	$129.08
7	Car	$40.45	$56.34	$139.78	$23.67	$25.34
8	Misc.	$140.56	$167.90	$239.78	$101.40	$407.34
9						
10	TOTAL	$380.45	$443.73	$591.52	$286.04	$622.63
11						
12						
13						
14						
15						
16						

Figure IV: A spreadsheet showing two panes

Database

Works allows a database consisting of up to 2,000 records (6,000 for a MacPlus) and each record can have up to 60 fields with 248 characters per field. The comparatively small number of records is a result of the whole database needing to be held in the Mac's ram.

Two views of the database are possible in Works, a Form View and a List View. It is easy to flip between the two by simply double clicking outside a data field with the mouse. The form view is ideal for entering new and editing old records since a complete record is shown in the window. The list view is better for skimming through the database or organising the data for producing reports.

Data can be entered as text, numeric, dates or times and treated accordingly. ▷

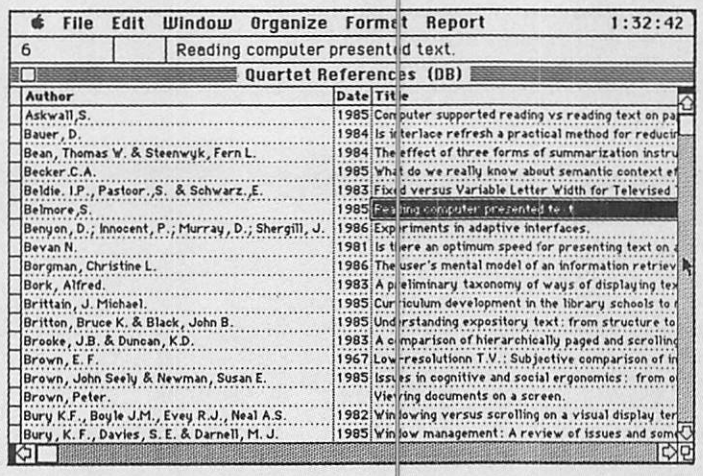


Figure V: A database file displayed in list format

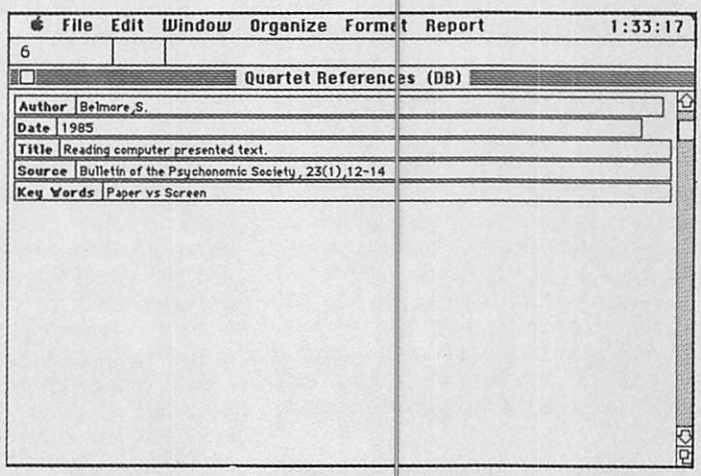


Figure VI: Displaying the database in form format

◁ Numeric fields can also be computed fields like those found in spreadsheets.

The number, size and position of fields within a record are easily changed - even after the database is well established. This is done on the form view by either rearranging the layout of fields with a 'grabber hand', MacPaint style, or by deleting old and inserting new fields.

There is a similar amount of manipulation possible with the list view of the database. The window can be divided into four independent panes in exactly the same way as the spreadsheet. The order of columns can be altered by dragging them with the mouse.

It is possible to supply information from the database to a Works word processor document using the Merge facility. This allows data such as names and addresses to be automatically incorporated into text documents – that's right, Works gives you the power to create your very own junk mail.

The database can be manipulated in a variety of ways – certainly enough to cover the majority of needs. It can be sorted alphabetically, numerically or chronologically, and in reverse order too. The database can be searched for the next occurrence of a particular string or for all the records containing a particular string.

Finally, a procedure with up to five rules of selection can be set up to produce reports from the database. The report can then be rearranged so that only a certain number of fields are shown for each record and the grid lines can be removed. In this way it is easy to produce tables to illustrate a document.

Communications

It is probable that the communications possibilities will be the least interesting aspect of Works for many users. But this situation may change radically in the future if BT ever manages to sort out its X.400 communications standard, and with it the possibilities for an integrated electronic mail system.

Secondly, although there are only a few online databases in this country, compared to the USA, if this service expands in a similar way the value of a comms package and modem could be rated much higher.

These points aside, what does Works offer? It is easy to learn and easy to use like the rest of the package. As with the other elements it provides a competent facility that caters for most every day needs, though auto-logon and a scrolling window would be welcome additions.

The package is based on the creation of communications documents each with their own settings for baud rate, data size, stop bits, parity and handshake. A phone book containing up to eight numbers can be linked to each document.

Works can handle data rates of between 300 and 19200 baud and is therefore suitable for use with most of the popular services, such as MicroLink, as well as fast data transfer between two local computers.

However, Works is an American product, so it cannot cope with Prestel's viewdata format or even split baud rates. There are facilities for capturing downloaded text and for sending and receiving files in MacBinary and X-modem text or data formats.

Integration

In general, the components that go to make up Works are all reasonably competent pieces of software, but how do they work together? The answer is very well – and the more you use Works the more you take the integration for granted. This is a telling sign.

When software is really well designed it becomes increasingly transparent and all you concern yourself with is getting on with the task at hand. It is when the software has inconsistent commands and complicated procedures to be remembered and executed when you transfer between modules that you start tripping over the joints.

Works allows up to 10 documents from any of the modules to be open on the desktop at once and you can cut and paste between any of them. This gives an indication of the sophistication of the supporting software that links the individual parts.

However, having 10 documents open at once on an 8" screen is obviously too many to handle without excessive shuffling. This point has been recognised by Microsoft since there is a pull-down menu which lists all of the Works documents currently open on the desk top.

However, this limitation disappears if you can run Works in conjunction with a big A3 screen like the MegaScreen. The two ideas seem to be made for each other since Works, unlike several other well established pieces of software, can make full use of the complete screen.

The space is a real bonus for spreadsheeting and charting as well as for switching between a number of opened documents. The scenario at the beginning of the article, which involved running all the components together, becomes a piece of cake.

Another very useful facility that helps you cope with a desktop covered with open documents is the Resume option. Whenever you start Works up you can select Resume and the desktop will be recreated exactly as when you last left it with all the documents opened at the right places.

Incidentally, the beta-test version of Works bombed repeatedly with the MegaScreen but the release version behaves perfectly. It's typical of Microsoft that such problems are sorted out by release date (even if that is delayed as a result). It's also

one of the reasons why *Apple User* prefers to review release versions rather than preview beta-test versions.

Now that MacWrite no longer comes bundled with the Mac there is a good case to be made for the new user to make Works his first software purchase. If you are an established Mac user, just think about all the time you could save not having to hang around while the Finder opens and closes applications.

The Works program takes up 327k on a disc and comes with a second disc containing a tutorial. Works runs quite happily on the old 512k Mac but obviously it is easier to use with 800k drives.

The discs are supported with a wealth of documentation comprising a 400 page manual, 150 page book of lessons and a quick reference guide. However, this is a classic case of belt and braces since the reasonably experienced Mac user should be able to get on top of the basics with little reference to the documentation.

At just under £300, Works represents good value for money.

Product: Microsoft's Works
Price: £281.75
Supplier: Microsoft, 49 de Montfort Road, Reading, Berkshire RG1 8LP.
Tel: 0734 500741

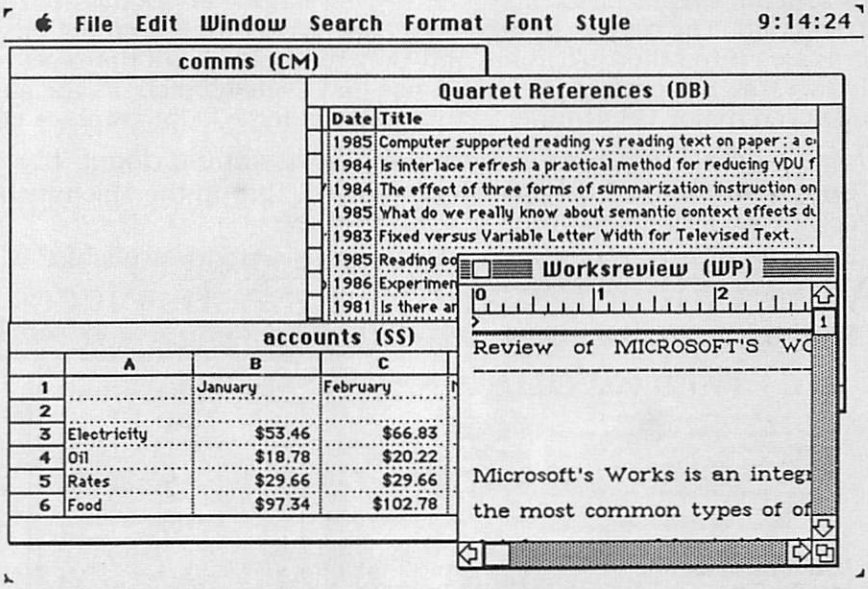


Figure VII: Up to 10 documents, with full cut-and-paste between them

The best of MacPaint and MacDraw:

SuperPaint is the most advanced graphics creation tool available for the Macintosh. It has two layers; one for editing dots like MacPaint, and one for manipulating objects like MacDraw. The features read like a Christmas wishlist. Full-screen editing. Multiple windows. 3 levels of magnification. Reduced view. Creates shapes bigger than the screen. Draws circles and squares from the centre. Text can be edited. LaserWriter Fonts. Open and save MacDraw PICT and MacPaint files. Print multiple copies. Colour printing on ImageWriter II. Makes full use of big screens.
Best of all, there's LaserBits™, dot-by-dot editing at 300dots-per-inch resolution. The results printed on a LaserWriter are stunning! Paste these graphics into other programs and they retain their 300dpi resolution!
In fact, the Superpaint file format has just been adopted as the standard by all seven major US scanner manufacturers for 300dpi graphics editing.

Here's what the US reviewers have been saying about it :

"**SuperPaint** is the best paint program available on the Macintosh today" - *Adrian Mello, MacWorld, Jan '87*
"I can really review **SuperPaint** in two words: Get it!" - *Sharon Aker, MacUser (US), Feb '87*
"**SuperPaint** is the hottest graphics package currently available." - *CJ Weigand, MACazine, Jan '87*

When all you want is a masterpiece SUPERPAINT £95.00

WriteNow

for Macintosh

- Document size only limited by disk space
- Number of documents open only limited by memory
- Instant repagination
- Visible on screen columns: 1 to 4
- 50,000 word built in spelling checker
- Imports MacWrite™, MS-Word™ and text files
- Visible pagebreaks
- Headers & footers with even/odd page specification
- Visible footnotes with auto page overflow
- Visible fonts from 4 to 127 points
- Graphic output embeddable in text
- Forward, backward, & wild card Find and Replace
- Sub- and Super- script in point increments
- Soft hyphenation
- Automatic backup file copies
- Alternating binding margins
- Tab indent left, indent right
- Left, centre, right, and arbitrary tabs
- "Keep on same Page" paragraphs
- Extended or condensed style options
- Date, time, and page no. anywhere on page
- Selective Font, size, and style changes
- "Identical paragraph" format changes
- Moveable ruler
- Fast, flicker-free, horizontal & vertical scrolling
- Cancel long operations mid event
- Option to show non printing characters
- Undo for ALL editing changes
- Not copy protected

... and it costs £165.00
Did we forget anything?

Introducing WriteNow For Macintosh, the next step in word processing for your Mac. WriteNow For Macintosh combines the power you would expect from a dedicated word processing system with the ease of operation that you're used to with MacWrite™
Word processing on the Mac will never be the same. Here's what Steve Jobs, creator of the Macintosh says about WriteNow:
"This is the word processor we designed and built the Macintosh for."
The wait is over. Right now.

The MacSerious Top 10
February 1987

1	SuperPaint
2	WriteNow
3	Dark Castle
4	MacGolf
5	Lightspeed Pascal
6	Lightspeed C
7	TML Pascal
8	More
9	Mac3D
10	Silicon Press

The MacSerious Promise

We're dedicated to helping the serious - or not too serious - Mac user get hold of the software he or she needs ... and which no-one else supplies this side of the Atlantic. If you can't find it elsewhere, try MacSerious. If we don't have it in stock, we'll get it for you if it exists.

MacSerious & the Apple //GS

Yes, we now have software for the GS. Available today:

Visualizer //GS (£95.00) Superb business graphics package taking full advantage of advanced features of the GS. Utilises AppleWorks data.

GraphicWriter (£140.00) Full powered WYSIWIG word-processor with built-in paint tools

PaintWorks Plus (£75.00) "Colour MacPaint" for the GS - and then some. With Paintworks Plus, you can make graphics come alive in animated sequences.

TML Pascal (APW) (£115.00, £140.00 w/ Source Code Library) Complete access to GS Tools; create stand alone applications.

Please Note: These products will not work on other Apple][machines.

This is just the beginning. At MacSerious we look forward to bringing you many more exciting new products for this superb reincarnation of the classic Apple][computer.

The MacSerious Programmers Toolbox

LightSpeed Pascal £115
An interactive compiler and development environment for the Mac. You'll find all the convenient debugging features of Macintosh Pascal (also from Think Tech) but the interactive program is seamlessly integrated with a high-performance compiler, ultra-fast linker, and automatic project management.
Lightspeed Pascal offers both the beginner and professional developer speed and ease of use in creating stand-alone double-clickable applications. Features include a High-Level Symbolic Debugger, Toolbox-level Debugger, numerous utilities, excellent documentation

LightSpeed C £160
A complete high-performance C programming environment providing in a single integrated Macintosh-style application, a multi-file text editor, high performance native code compiler, ultra-fast linker, and automatic make facility, as well as full ToolBox and Unix-compatability library support. Complete implementation of the C language as defined by Kernighan & Ritchie's *The C Programming Language* plus more recent features.
Lightspeed C compiles more than 10 times faster than any other Mac C compiler. Generated code is between 70% & 90% of the size produced by other compilers, and execution time varies between 65% and 95%. Call for full details.

MORE

The First Integrated Idea Processor/Idea Presenter
More is a third generation idea processing software product from Living Videotext, the company that invented Macintosh idea processing with Thinktank
Software Product of the Year 1986, MacUser Magazine (USA)

- **MORE Intelligent Idea Processing**
  - Hoisting
  - Cloning
  - Automated Reorganization
  - Pattern Matching
  - Visual Levels
- **MORE Desktop Presentations**
  - Bullet Charts
  - Tree Charts
  - Direct Transfer to DTP Programs
- **MORE Desktop Productivity**
  - Outline Templates
  - Smart Calendars and Time Stamping
  - Auto Phone Dialer
  - Outline Maths
  - Text and Graphics Windows
- **MORE Macintosh Power**
  - Standard editing
  - Up To Six Windows Open
  - Window Tiling
  - Transfer Outlines to Text and PICT

£295.00

Trapeze

The Spreadsheet without limits
Trapeze is a revolutionary program that frees you from the limitations of row and column spreadsheets. It was designed specifically for the Mac and takes full advantage of it's speed, power and simplicity.

Here's what Trapeze does...
Trapeze stores information in blocks; it does not lock you into a row & block prison. And you can move them freely around wherever you want, retaining the relationship between blocks.

And it does it Fast...
Complex worksheets can be set up in minutes. Trapeze has fast, superious calulation speed, and it directly supports Levco's Prodigy 4™ and other 68881 maths coprocessors.

Power on Demand...
Up to 32 worksheets open at once.)ver 100 built-in functions. Create charts with 1000's of plots. And model size is limited only by memory.

Plus Things No Other Spreadsheet Can Do...
Advanced operations, such as desktop engineering functions ranging from matrix operations to simultaneous equations. Multiple databases for efficient organization, analysis and presentation. Import graphics and text from paint, draw and word processing programs.

Trapeze costs £295.00

From Jasik Designs

The interactive disassembler.
When you've got MacNosy, there are no more secrets.
£80.00

The Debugger
beyond Discipline.. the ultimate one
£150.00

TML Pascal £90
We could be selling it for 2 or 3 times the price. This amazing system will compile BOTH Lisa & Macintosh Pascal, permitting the creation of stand-alone, double-clickable applications, and DA's.
And to go with it there's TML Systems' Development Tools....
TML Source Code Library for only £70.00: A collection of 18 example Pascal applications rep-resenting over 1 Mb of source code.
MacLanguage Database Toolkit for £85.00 - a complete library of Pascal procedures.
MacExpress for Pascal from AlSoft for £175.00 is a complete library of Pascal procedures.

The people who know Mac software

MacSerious

The MacSerious Company
17 Park Circus Place
Glasgow G3 6AH

Phone 041-332-5622 Fax 041-332-3209 Telex 777021 MACSER G

MacSerious is *the* source for Mac (& GS) software in Britain and Europe. Write or phone for our free 550+ item catalogue today!

In Europe, call us on these numbers:

Germany:	06583-1371
Netherlands:	080-239520
Irish Republic:	01-987-001
Switzerland:	01-59-11-77
Belgium:	091-85-55-11
Spain:	01-262-4113

DEALERS AND DISTRIBUTORS
CALL FOR TRADE INFORMATION
Let us help you get the products your customers want.

All offers subject to availability. We reserve the right to adjust prices without notice.

A little plastic notice
We take orders paid with Access, American Express, EuroCard, MasterCard and Visa

Shipping & Handling:
In the UK: £2.00/order (1st class mail)
W. Europe: £3.50/order (normal mail)
Call for costs by other ways or to other places.

PRICES SHOWN HERE DO NOT INCLUDE VAT

Updating Dos disc sectors

Lee Hammond's utility program takes the tedium out of working with Dos disc sectors

THIS utility program, written almost entirely in Basic, provides a simple and effective method of reading, printing and updating Dos disc sectors.

Facilities I have included are sector printing, comparing, copying, editing, as well as the disc verify routine by D. Poirier (in the May 1985 issue of *Apple User*.)

Each facility is written as a standalone piece of code, supported by several common subroutines which make the creation of new facilities quite simple.

PRINT: Will read a sector from disc and dump it to the printer. Data is formatted into a hexadecimal and Ascii dump. Figure I gives an example of the output produced. If you have an 80 column card you may find it useful to change line 2170 to PR#3 and to insert line 2205 GET ZZ$.

COMPARE: Will compare two disc sectors and produce a report of any differences to the printer. Figure IIa shows the output produced when both sectors are identical. Figure IIb shows the type of report produced when sectors differ. Any differing bytes are highlighted with an * for Ascii and ** for hexadecimal.

COPY: Will copy one sector to another, either on the same disc or between two.

VERIFY: Is a copy of D. Poirier's verify routine from *Apple User* May '85.

EDIT: Allows you to read a sector from disc, update it on-screen and write it back to the disc.

When displayed on-screen, the sector is displayed as two "pages": Page One contains bytes 00-7F and Page Two bytes 80-FF. Page switching is accomplished by entering P.

The data within a sector is updated simply by moving the **[]** cursor across the screen and overkeying the required byte with valid hexadecimal characters.

The cursor is moved by pressing either the left/right arrow keys or the spacebar. Whenever the cursor moves beyond the end of a page it will reappear at the start or end of the same page, depending on the direction of movement.

The editor is terminated by the Escape key. If the sector has been updated, you are prompted to see if it should be written back to disc. See Figures IIIa and IIIb for screen dumps from the editor.

```
    VOL:254     TRACK:17     SECTOR:15


+00 - 00 11 0E 00 00 00 00 00 00 00 12 0F B2 C8 C5    ..............HE
+10 - CC CC CF A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0    LLO
+20 - A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 02 00 1B 0F            ....
+30 - 02 C4 C9 D3 CB B3 AE C2 C1 D3 C9 C3 A0 A0 A0 A0    .DISK3.BASIC
+40 - A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 12                .
+50 - 00 FF 0F 02 D2 C1 C3 C5 AE C3 D2 C5 C1 D4 C5 A0    ....RACE.CREATE
+60 - A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0
+70 - A0 1B 03 00 FF 0F 02 D2 C1 C3 C5 AE C4 C9 D3 D0     ......RACE.DISP
+80 - CC C1 D9 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0    LAY
+90 - A0 A0 A0 A0 19 03 00 FF 0F 02 D2 C1 C3 C5 D2 D3        ......RACERS
+A0 - A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0
+B0 - A0 A0 A0 A0 A0 A0 A0 1A 08 00 FF 0F 00 D4 D7 CF              ......TWO
+C0 - AE C3 CF D5 D2 D3 C5 A0 A0 A0 A0 A0 A0 A0 A0 A0    .COURSE
+D0 - A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 1B 08 00 FF 0F 02                ......
+E0 - C5 C4 C9 D4 C9 CE C7 A0 D4 D5 D4 CF D2 A0 A0 A0    EDITING TUTOR
+F0 - A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 A0 1C 0D 00                   ...
```

Figure I: Disc sector print facility

```
DEVICE #1    VOL:254    TRACK:17    SECTOR:15
DEVICE #2    VOL:254    TRACK:17    SECTOR:15

        0 DIFFERENCES DETECTED
```

Figure IIa: Disc sector compare facility

```
DEVICE #1    VOL:254    TRACK:17    SECTOR:15
DEVICE #2    VOL:254    TRACK:17    SECTOR:14

DEV#1  +00 - 00 11 0E 00 00 00 00 00 00 00 00 12 0F B2 C8 C5    ..............HE
             ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** **    **************
DEV#2  +00 - C5 C5 C5 C5 C5 C5 C5 C5 C5 C5 C5 C5 C5 C5 C5 C5    EEEEEEEEEEEEEEEE

      15 DIFFERENCES DETECTED
```

Figure IIb: Disc sector compare facility

```
+00  [D5] D5  D5  D5  D5  D5  D5  D5
+08   D5  D5  D5  D5  D5  D5  DD  D5
+10   CC  CC  CF  A0  A0  A0  A0  A0
+18   A0  A0  A0  A0  A0  A0  A0  A0
+20   A0  A0  A0  A0  A0  A0  A0  A0
+28   A0  A0  A0  A0  02  00  1B  0F
+30   02  C4  C9  D3  CB  B3  AE  C2
+38   C1  D3  C9  C3  A0  A0  A0  A0
+40   A0  A0  A0  A0  A0  A0  A0  A0
+48   A0  A0  A0  A0  A0  A0  A0  12
+50   00  FF  0F  02  D2  C1  C3  C5
+58   AE  C3  D2  C5  C1  D4  C5  A0
+60   CC  CC  CC  CC  CC  CC  CC  CC
+68   CC  CC  CC  CC  CC  CC  CC  CC
+70   CC  CC  CC  CC  CC  CC  CC  CC
+78   DD  DD  DD  DD  DD  DD  DD  DD

  P NEXT PAGE     < PREV      > NEXT

ESC END EDIT
```

Figure IIIa: Sector edit facility

```
+80   CC  C1  D9  A0  A0  A0  A0  A0
+88   A0  A0  A0  A0  A0  A0  A0  A0
+90   A0  A0  A0  A0  19  03  00  FF
+98   0F  02  D2  C1  C3  C5  D2  D3
+A0   A0  A0  A0  A0  A0  A0  A0  A0
+A8   A0  A0  A0  A0  A0  A0  A0  A0
+B0   A0  A0  A0  A0  A0  A0  A0  1A
+B8   08  00  FF  0F  00  D4  D7  CF
+C0   AE  C3  CF  D5  D2  D3  C5  A0
+C8   A0  A0  A0  A0  A0  A0  A0  A0
+D0   A0  A0  A0  A0  A0  A0  A0  A0
+D8   A0  A0  1B  08  00  FF  0F  02
+E0   C5 [C4] C9  D4  C9  CE  C7  A0
+E8   D4  D5  D4  CF  D2  A0  A0  A0
+F0   A0  A0  A0  A0  A0  A0  A0  A0
+F8   A0  A0  A0  A0  A0  1C  0D  00

  P PREV PAGE     < PREV      > NEXT

ESC END EDIT
```

Figure IIIb: Sector edit facility

SUBROUTINES:

Line #

9000 Converts decimal number (0-255) supplied in Z into two character hexadecimal string in X$.

9010 Returns disc slot and drive numbers in S and D.

9040 Returns disc track number in T.

9060 Returns disc sector number in S.

9080 Using B as pointer to disc sector, and I as index into buffer, print 16 bytes in hexadecimal.

9090 Using B as pointer to disc sector, and I as index into buffer, print 16 bytes in Ascii. Note that I have assumed the high bits to be set for Drs. Other systems may need the bit reset in which case compare with 32 and 127.

9120 Call RWTS using parameters in arrays:
B() – Buffer pointers
S() – slot numbers
D() – Drive numbers
T() – Track numbers
(L) – Sector numbers.
Z is set to 1 or 2 for element selection. Z9 is set to 1 if error checking is not required.

```
1000 REM  * DISK UTILITY *
1010 OH =  PEEK (115)
     +  PEEK (116) * 256
   :NH = OH - 1024
   : HIMEM: NH
1020 DIM M(15)
1030 MAX = 5
1040 D$ =  CHR$ (4)
1050 B(1) =  PEEK (116)
     + 1
   :B(3) = B(1) * 256
1060 B(2) = B(1) + 1
   :B(4) = B(2) * 256
1070 H$ = "0123456789ABCDEF"
1080 A$(1) = "INPUT"
   :A$(2) = "OUTPUT"
1090 A$(3) = "NEXT"
   :A$(4) = "PREV"
   :A$(5) = "DISK WRITE PRO
     TECTED"
   :A$(6) = "VOLUME MISMATC
     H"
   :A$(7) = "DRIVE ERROR"
   :A$(9) = "READ ERROR"
1100 FOR I = 769 TO 812
     : READ A
     : POKE I,A
     : NEXT
1110 DATA 1,0,0,0,0,0,18,3
1120 DATA 0,96,0,0,1,0,0,0,0
1130 DATA 0,1,239,216
1140 DATA 160,1,169,3,32,217
     ,3,176,9,160
1150 DATA 0,132,254,160,0,13
     2,72,96
1160 DATA 160,255,76,33,3
1170 PRINT D$"PR£1"
   : POKE 1657,80
1180 PRINT D$;"PR£0"
1190 M$ = ""
2000 HOME
   : PRINT "DISK UTILITY"
2010 POKE 781,1
   :Z9 = 0
2020 VTAB 3
   : HTAB 1
   : PRINT "  1 - PRINT DISK
     SECTOR"
2030 PRINT "  2 - COMPARE DI
     SK SECTORS"
2040 PRINT "  3 - COPY DISK
     SECTORS"
2050 PRINT "  4 - SECTOR EDI
     TOR"
2060 PRINT "  5 - DISKETTE V
     ERIFY"
2070 VTAB 18
   : HTAB 1
   : PRINT "  0 - QUIT"
2080 VTAB 20
   : PRINT M$
   : VTAB 22
   : HTAB 1
   : INPUT "ENTER OPTION
   : ";OP
2090 IF OP <  >  INT (OP)
     OR OP < 0 OR OP > MAX
     THEN M$ = "INVALID OPTI
     ON ENTERED" +  CHR$ (7)

   : GOTO 2080
2100 IF OP = 0 THEN  HOME
   : HIMEM: OH
   : END
2110 IF OP = 2 THEN 2230
2120 IF OP = 3 THEN 2440
2130 IF OP = 4 THEN 2510
2140 IF OP = 5 THEN 2870
2150 B = B(3)
   : HOME
   : PRINT "SECTOR PRINT FAC
     ILITY"
   : PRINT
   : PRINT
2160 GOSUB 9010
   :S(1) = S
   :D(1) = D
   : PRINT
   : GOSUB 9040
   : GOSUB 9060
   :T(1) = T
   :L(1) = S
   :Z = 1
   : GOSUB 9120
2170 PRINT D$;"PR£1"
2180 PRINT "DISK SECTOR PRIN
     T FACILITY"
   : PRINT
   : PRINT
2190 PRINT "  VOL
   :" PEEK (783)"    TRACK
   :"T"    SECTOR
   :"S
   : PRINT
   : PRINT
2200 FOR I = 0 TO 15
     : PRINT
     : PRINT "+":
     :Z = I * 16
     : GOSUB 9000
     : PRINT X$;" - ";
     : GOSUB 9080
     : PRINT "   ";
     : GOSUB 9090
     : NEXT
2210 M$ = "PRINT COMPLETED"
   : PRINT
   : PRINT D$;"PR£0"
2220 GOTO 2000
2230 HOME
   : PRINT "SECTOR COMPARE F
     ACILITY"
   : PRINT
2240 FOR I = 1 TO 2
     : PRINT
     : PRINT "ENTER DETAILS
       FOR DEVICE £"I
     : GOSUB 9010
     :S(I) = S
     :D(I) = D
     : PRINT
     : GOSUB 9040
     : GOSUB 9060
     :L(I) = S
     :T(I) = T
     : PRINT
     : PRINT
     : NEXT
2250 FOR I = 1 TO 2
     :Z = I
     : GOSUB 9120
2260   V(I) =  PEEK (783)
2270   NEXT
2280 PRINT D$;"PR£1"
   : PRINT "DISK SECTOR COMP
     ARE FACILITY"
   : PRINT
   : PRINT
   : FOR I = 1 TO 2
     : PRINT "DEVICE £"I"
       VOL
     :"V(I)"    TRACK
     :"T(I)"    SECTOR
     :"L(I)
2290   NEXT
2300 M = 0
   : FOR I = 0 TO 15
```

```
     : FOR J = 0 TO 15
       :M(J) = 0
       : NEXT
     :M2 = 0
     : FOR J = 0 TO 15
2310     IF  PEEK (B(3)
         + I * 16 + J)
         <  >  PEEK (B(4)
         + I * 16 + J)
         THEN M2 = M2
         + 1
       :M(J) = 1
       :M = M + 1
2320     NEXT
     : IF M2 = 0 THEN 2410
2330   PRINT
     : PRINT "DEV£1  +";
     :Z = I * 16
     : GOSUB 9000
     : PRINT X$;" - ";
     :B = B(3)
     : GOSUB 9080
     : PRINT "    ";
     : GOSUB 9090
     : PRINT
2340   PRINT  SPC( 13);
     : FOR J = 0 TO 15
       : IF M(J) = 0 THEN
         PRINT "    ";
2350     IF M(J) = 1 THEN
         PRINT "** ";
2360     NEXT
2370   PRINT "    ";
     : FOR J = 0 TO 15
       : IF M(J) = 0 THEN
         PRINT " ";
2380     IF M(J) = 1 THEN
         PRINT "*";
2390     NEXT
2400   PRINT
     : PRINT "DEV£2  +";
     :Z = I * 16
     : GOSUB 9000
     : PRINT X$;" - ";
     :B = B(4)
     : GOSUB 9080
     : PRINT "    ";
     : GOSUB 9090
     : PRINT
2410   NEXT
2420 PRINT
   : PRINT
   : PRINT  SPC( 6):M:" DIFF
     ERENCES DETECTED"
   : PRINT D$;"PR£0"
2430 M$ = "COMPARE COMPLETED
      - " +  STR$ (M)
     + " DIFFERENCES"
   : GOTO 2000
2440 HOME
   : PRINT "SECTOR COPY FACI
     LITY"
   : PRINT
   : PRINT
2450 FOR I = 1 TO 2
     : PRINT "ENTER DETAILS
       FOR "A$(I)" DEVICE"
2460   GOSUB 9010
     :S(I) = S
     :D(I) = D
     : PRINT
     : GOSUB 9040
     : GOSUB 9060
     :T(I) = T
     :L(I) = S
     : PRINT
     : PRINT
     : NEXT
2470 PRINT
   : PRINT "ENTER 'Y' TO CON
     FIRM WRITE"
   : PRINT "TO SLOT "S(2)".
     DRIVE "D(2)"  ";
2480 GET A$
   : IF A$ <  > "Y" THEN M$
     = "COPY TERMINATED"
   : GOTO 2000
2490 FOR I = 1 TO 2
     : POKE 781,I
     :Z = I
     : GOSUB 9130
     : NEXT
   : POKE 781,1
2500 M$ = "COPY COMPLETED"
   : GOTO 2000
2510 HOME
   : PRINT "SECTOR EDIT FACI
     LITY"
   : PRINT
   : PRINT
2520 GOSUB 9010
   :S(1) = S
   :D(1) = D
   : PRINT
   : GOSUB 9040
   : GOSUB 9060
   :T(1) = T
   :L(1) = S
   :Z = 1
   : GOSUB 9120
2530 HOME
   : PRINT "SECTOR EDIT FACI
     LITY"
   : PRINT
   : PRINT
2540 P = 0
   :O = 0
   :C = 6
   :R = 4
   :U = 0
   :O1 = 0
2550 FOR I = P * 128 TO P
       * 128 + 127 STEP 8
     :Z = I
     : GOSUB 9000
     : PRINT "+"X$"   ";
2560   FOR J = 0 TO 7
       :Z =  PEEK (B(3)
         + I + J)
       : GOSUB 9000
       : PRINT X$"  ";
       : NEXT
     : PRINT
2570   NEXT
2580 VTAB 21
   : HTAB 3
   : INVERSE
   : PRINT "P";
   : NORMAL
   : PRINT " "A$(P + 3)" PAG
     E    ";
   : INVERSE
   : PRINT "<";
   : NORMAL
   : PRINT " PREV     ";
   : INVERSE
   : PRINT ">";
   : NORMAL
   : PRINT " NEXT"
2590 PRINT
   : INVERSE
   : PRINT "ESC";
   : NORMAL
   : PRINT " END EDIT";
2600 GOSUB 9190
2610 VTAB R
   : HTAB C + O1 + 1
2620 GET M$
   :M =  ASC (M$)
2630 IF M = 27 THEN 2820
2640 IF M$ = "P" THEN
     GOSUB 9180
   :P =  NOT (P)
   :R = 4
   :C = 6
   :O = P * 128
   : VTAB 4
   : HTAB 1
   : GOTO 2550
2650 VTAB R
   : HTAB C + O1 + 1
2660 IF M = 27 THEN 2820
2670 IF M = 32 THEN M
     = 21
2680 IF M = 8 THEN  GOSUB 27
     60
   :O1 = 0
   : GOTO 2610
2690 IF M = 21 THEN  GOSUB 2
     790
   :O1 = 0
   : GOTO 2610
2700 FOR J = 1 TO 16
     : IF  MID$ (H$,J,1)
       <  > M$ THEN  NEXT
   : GOTO 2610
2710 Z = J
   :J = 17
   : NEXT
2720 PRINT M$;
2730X =  PEEK (B(3) + O)
 :X1 =  INT (X / 16)
 :X2 = X - (X1 * 16)
 : IF O1 = 0 THEN X1 = Z
   - 1
2740 IF O1 = 1 THEN X2 = Z
   - 1
2750 POKE B(3) + O,X1 * 16
   + X2
 :U = 1
 :O1 =  NOT (O1)
```

```
 : GOTO 2610
27600 = O - 1
 : GOSUB 9180
 : IF C <  > 6 THEN C
   = C - 4
 : GOSUB 9190
 : RETURN
2770R = R - 1
 :C = 34
 : IF R < 4 THEN R = 19
 :O = P * 128 + 127
 : GOSUB 9190
 : RETURN
2780 GOSUB 9190
 : RETURN
27900 = O + 1
 : GOSUB 9180
 : IF C <  > 34 THEN C
   = C + 4
 : GOSUB 9190
 : RETURN
2800R = R + 1
 :C = 6
 : IF R > 19 THEN R = 4
 :O = P * 128
 : GOSUB 9190
 : RETURN
2810 GOSUB 9190
 : RETURN
2820 IF U = 0 THEN M$ = "EDIT
   COMPLETED - NO UPDATES"
 : GOTO 2000
2830 VTAB 21
 : HTAB 1
 : PRINT " ";
 : CALL  - 958
2840 VTAB 21
 : HTAB 1
 : INPUT "WRITE SECTOR BACK
   TO DISK ? ";M$
2850 IF  LEFT$ (M$,1) <
   > "Y" THEN M$ = "EDIT TER
   MINATED"
 : GOTO 2000
2860 POKE 781,2
 :Z = 1
 : GOSUB 9120
 :M$ = "EDIT COMPLETED"
 : GOTO 2000
2870 HOME
 : PRINT "DISKETTE VERIFY FA
   CILITY"
 : PRINT
 : PRINT
2880 GOSUB 9010
 :S(1) = S
 :D(1) = D
 :Z9 = 1
 :M = 0
2890 HOME
2900 VTAB 3
 : HTAB 3
 : PRINT "                  11
   1111111111111222"
 : PRINT "  0123456789ABCDEF
   0123456789ABCDEF012"
2910 FOR L = 0 TO 9
   : VTAB L + 5
   : HTAB 2
   : PRINT L
   : NEXT
 : PRINT " A"
 : PRINT " B"
 : PRINT " C"
 : PRINT " D"
 : PRINT " E"
 : PRINT " F"
2920 INVERSE
2930 VTAB 2
 : HTAB 18
 : PRINT "TRACK"
 : VTAB 10
 : PRINT "S"
 : PRINT "E"
 : PRINT "C"
 : PRINT "T"
 : PRINT "O"
 : PRINT "R"
2940 NORMAL
2950 FOR T = 0 TO 34
   :T(1) = T
   : FOR S = 0 TO 15
     :L(1) = S
     :Z = 1
     : GOSUB 9120
     : HTAB T + 3
     : VTAB S + 5
     :T$ = "."
     : IF  PEEK (254)
       <  > 0 THEN T$
       = "*"
     :M = M + 1
2960   PRINT T$
     : NEXT
   : NEXT
2970 VTAB 23
 : HTAB 1
 : PRINT "PRESS ANY KEY TO C
   ONTINUE ";
 : GET T$
2980M$ = "VERIFY COMPLETE - "
   +  STR$ (M) + " ERRORS"
 : GOTO 2000
8990 END
9000Z1 =  INT (Z / 16)
 :Z2 = Z - Z1 * 16
 :X$ =  MID$ (H$,Z1 + 1,1)
   +  MID$ (H$,Z2 + 1,1)
 :Z =  FRE (0)
 : RETURN
9010 INPUT "SLOT
 : ";S
 : IF S <  >  INT (S)
   OR S < 3 OR S > 7 THEN 90
   10
9020 INPUT "DRIVE
 : ";D
 : IF D <  >  INT (D)
   OR D < 1 OR D > 2 THEN 90
   20
9030 RETURN
9040 INPUT "TRACK
 : ";T
 : IF T <  >  INT (T)
   OR T < 0 OR T > 34
   THEN 9040
9050 RETURN
9060 INPUT "SECTOR
 : ";S
 : IF S <  >  INT (S)
   OR S < 0 OR S > 15
   THEN 9060
9070 RETURN
9080 FOR Z3 = 0 TO 15
   :Z =  PEEK (B + I
     * 16 + Z3)
   : GOSUB 9000
   : PRINT X$;" ";
   : NEXT
 : RETURN
9090 FOR Z3 = 0 TO 15
   :Z =  PEEK (B + I
     * 16 + Z3)
   : IF Z < 160 OR Z >
     = 224 THEN  PRINT ".";
   : GOTO 9110
9100 PRINT  CHR$ (Z);
9110 NEXT Z3
 : RETURN
9120 POKE 778,B(Z)
9130 POKE 770,S(Z) * 16
 : POKE 784,S(Z) * 16
 : POKE 771,D(Z)
 : POKE 785,D(Z)
 : POKE 773,T(Z)
 : POKE 774,L(Z)
 : CALL 790
 : IF  PEEK (254) = 0
   THEN  RETURN
9140 IF Z9 = 1 THEN  RETURN
9150 POP
9160 PRINT D$;"PR£0"
 :M$ = "DISK ERROR £"
   +  STR$ ( PEEK (782))
   + " ("
9170M$ = M$ + A$( INT (
   PEEK (782) / 32) + 5)
   + ")" +  CHR$ (7)
 : GOTO 2000
9180 VTAB R
 : HTAB C
 : PRINT " ";
 : HTAB C + 3
 : PRINT " ";
 : RETURN
9190 VTAB R
 : HTAB C
 : PRINT  CHR$ (219);
 : HTAB C + 3
 : PRINT  CHR$ (221);
 : RETURN
9200Z = 0
 : FOR Z1 = 1 TO 16
   : IF  MID$ (H$,Z,1)
     = Z$ THEN Z = Z1
     - 1
   :Z = 17
   : NEXT
 : RETURN
9210 NEXT
```

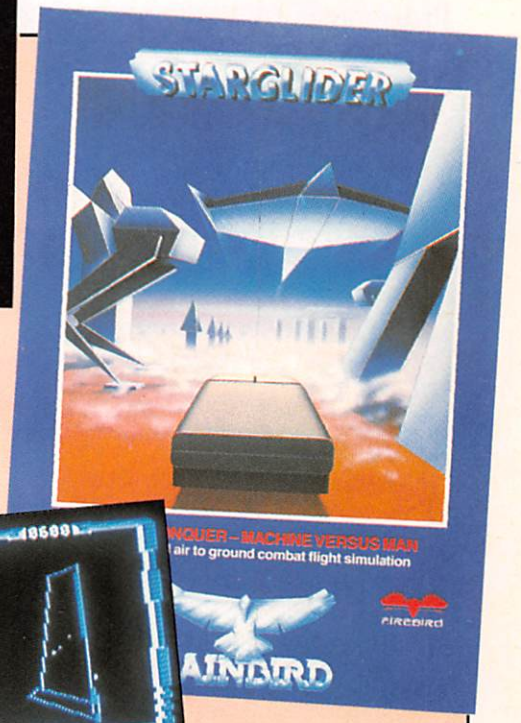
AppleUser SPECIAL OFFERS!

Now on the Apple II – the game that took the software world by storm!

STARGLIDER is an all-action flight simulation with the perfect mixture of strategy and dexterity.

Swoop round the towers and blast the stompers. Seek out and destroy the menacing Starglider One with the flapping wings and dock with the rotating missile silo.

A masterpiece of programming, design and execution!



✔VOTED **Best Arcade Game on the Atari ST**

Suitable for	R.R.P.	Special Reader offer	YOU SAVE	Offer including subscription	YOU SAVE
Apple IIe	£19.95	£14.95	£5	£27.95	£10

SAVE UP TO £10

This is the classic game no Apple user should miss . . . now it comes to you at an exceptionally low price!

Elite challenges you to undertake a fantastic voyage of discovery and adventure – making it a supreme test of your combat, navigational and entrepreneurial skills.

There are more than 2,000 different planets you can visit, bartering with their inhabitants, fighting off space pirates and bounty hunters.

The package includes a Space Trader's Flight Training Manual, a short novel to set the scene, a Quick Key Control guide, a Ship Identification guide – all designed to help you make the most of this superb game.

Suitable for	R.R.P.	Special Reader offer	YOU SAVE	Offer including subscription	YOU SAVE
Apple IIe	£19.95	£14.95	£5	£27.95	£10

✔VOTED **America's No. 1 Apple game**

TO ORDER PLEASE USE THE FORM ON PAGE 73

Flipping around AppleWorks

Jaromir Smejc teaches an old workhorse a few new tricks

SEVERAL years ago, working with my old Apple II+, I started to use AppleWriter. As soon as it was available I switched to the IIe and AppleWriter II, and around this time I started my database with Quick File.

Later on I used AppleWorks, but only for its database (it's more powerful than Quick File and I could transfer my old files into it) and simple speadsheet files. For my word processing work I still used AppleWriter II – and the reasons for this are simple:

- I had only 128k of memory, and after adding database files to the desktop there was very little space available – certainly not enough for word processing files, and anyway I had to change the discs.
- AppleWorks' inability to send all the printer codes I needed. Especially, codes for backspace, an international character set and printing different sizes and character styles (such as italics and NLQ) in a single line.

AppleWorks also lacks 7/72in line spacing and downloadable fonts, both essential for printing a graphic in a document with the aid of a specially designed downloadable font.

And AppleWorks allows only four functions to be used in one line: Bold, underline, subscript and superscript.

All other printer control codes – for example for different character sizes – can be used only for whole paragraphs.

My venerable Apple Writer II has no such limitations and allows me to send all printer control codes directly from within a document. This includes such commands as disable and enable (to not print parts of a document), and non-printable one line remarks starting with . as well.

This was my situation before Flipper. With a Flipper card installed I had enough memory – all that remained was to remedy AppleWork's deficiencies to have an ideal combination.

Possibilities included quick and simple transfer of spreadsheets or database files to word processor files, and a broad range of accessories such as calendars, outliners, macros and a communications ability. Or I could use the complete Megahaus and Pinpoint add-ons.

The advantages of such an integrated system are obvious and thanks to the Flipper, work is accomplished faster too. And with version 2.0 of AppleWorks I should be able to use more characters per record (and still have 30 categories) in my database files, new functions in spreadsheet and so on, but with the printer control codes unchanged.

Gaining control of all the printer's features can make documents more expressive – different typestyles, typefaces and sizes to highlight key points and so on. And if you want, as I do, to use headers, footers and paragraph names for various documents, mostly as a kind of template, control of your printer's abilities is a must.

The AppleWorks manual recommends a very unsatisfactory, limited remedy for a small part of the problem. If you need other control codes, you have to enter the proper ones under the boldface (or subscript, superscript or underline) option.

Thus, if you typed in the command for boldface (Control+B) your printer would print with the feature you turned on with the control codes entered under the boldface option name – but you lose the ability to print in bold. Besides, you have only four different control codes at your disposal and I found this limitation unacceptable. In some documents I need different typestyles, faces, sizes and special symbols in a single line – not very often, but often enough to look for a better solution than that recommended in the AppleWorks manual.

Therefore I've experimented with AppleWorks. And I've found the answer lies in its ability to accept the Escape code (without further character or control code), and in the firmware in the Epson FX-85 printer.

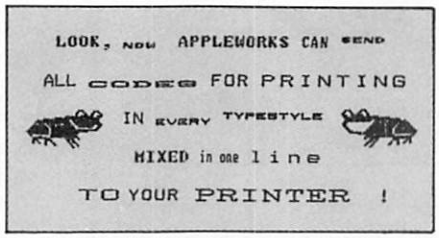


Figure I: Possibilities resulting from printer code control from AppleWorks

If the characters following the Escape code are part of a printer control code, they are not printed, but executed as a command. For example, in an Applesoft program the line:

```
100 PRINT CHR$ (27);'MAPPLE'
```

will put MAPPLE on the screen. But Epson FX,JX and EX printers (and many others as well) would print APPLE in Elite typesize instead of the normal Pica – the firmware translates the sequence ESC MAPPLE into the command ESC M.

With this in mind, and after many hours testing, I came up with the solution presented here. I find it surprisingly effective, with features that make AppleWorks even better.

So if you have AppleWorks and a printer using Escape sequences as control codes – that is, Esc code CHR$ (27) followed by any character – you will be able to add more flexibility to your printouts.

You will now be able to:

- Use all normal AppleWorks functions and commands.
- Mix any typefaces (custom download font, NLQ font) and typestyles (italics, bold and so on).
- Mix different spacing in proportional mode – normally not accessible via the custom printer driver.
- Mix type sizes within one or several lines, paragraphs and whole documents and print them in every configuration your printer allows. More than 80 font sizes, styles and modes (not counting download font, proportional and underlined varieties) are possible.
- Print graphics as part of your document, using the custom downloadable font.
- Use previously inaccessible printer control codes, because there are no word processing commands in AppleWorks that will let you choose printing in Enhanced mode. (Bold mode is AppleWorks' name for Epson's double-strike mode.)
- Use backspace, deselect and again select the paper–end detector: To select and deselect the printer (useful for non-printable remarks anywhere in your document), to print only from leftmost to right, to use line spacing other than 6 or 8 lines per inch and so on.

This will work on any Epson FX, JX, EX

and many non-Epsons as well, including Apple's DMP and Imagewriter I and II.

- Set up a custom printer driver providing all these abilities as a permanent solution.

However, note that all the following recommendations are based on my work with an Epson FX-85: If you have another model you may have to omit or adapt some commands.

And please read the whole article before creating the necessary custom printer driver: The explanations for the new codes and how they work are not directly in the instructions but later in the article. If you have to change some codes you'll need to understand why and how these commands work.

The commands have been thoroughly tested for more than three months with AppleWorks version 1.3 modified for the Flipper card and for Beagle Bros' MacroWorks. I've used an Epson FX-85 Version 3.0 on an Apple IIe with Epson parallel card (#8132 upgraded with the Dark Star Eprom ImageMaker EPCI).

The following instructions are for creating your own custom printer driver file: When creating it you have to enter the commands which correspond to the limited special printer functions (such as bold, underlined, characters per inch and so on) as well as our new ones. Here is the procedure:

- Boot the AppleWorks startup and program discs as usual.
- From the main menu choose option 5, other activities
- Choose option 7, specify information about your printer(s).
- Choose option 2, add a printer.
- Choose option 12, custom printer: (note that you can only have one custom printer)
- Type a name for the printer – choose a different one from the one you have as option 1.

```
CHARS.   REAL CONTENTS
PER      + = function Begin                         ENTER
INCH     - = function End
.......................................................................

 4       none (RESERVE)                             nothing
 5       5 chars. per inch                          Escape ! Space ^
 6       6 chars. per inch                          Escape ! ! ^
 7       none (RESERVE)                             nothing
 8       8 chars. per inch                          Escape ! $ ^
 9       9 chars. per inch                          Escape ! % ^
10       10 chars. per inch                         Escape ! Control-@ ^
11       none (RESERVE)                             nothing
12       12 chars. per inch                         Escape ! Control-A ^
13       NLQ font  +                                Escape x 1 ^
14       NLQ font  -                                Escape x Ø ^
15       Download font  +                           Escape % Control-A Control-@ ^
16       Download font  -                           Escape % Control-@ Control-@ ^
17       17 chars. per inch                         Escape ! Control-D ^
18       Sweden chars. set  +                       Escape R Control-E ^
19       Sweden chars. set  -                       Escape R Control-@ ^
20       20 chars. per inch                         Escape ! Control-E ^
21       20 chars. per inch & Superscript  +        Escape ! Control-D Escape S Ø ^
22       20 chars. per inch & Superscript  -        Escape T Escape ! Control-@ ^
23       Select printer                             Control-Q ^
24       Deselect printer                           Control-S ^
```

Figure II: Possible commands for characters per inch

- Repeat your original printer specification in the first four items for the custom printer.
- Choose option 5, interface card, and be careful not to accept the default CTRL-I 80N command – otherwise you will be unable to print more than 80 characters on a line.

First consult your interface card manual: You can try to use the **none** code, CTRL-I 0N, CTRL-I 137N (for eight inch platen printers) or CTRL-I 255N. Serial cards may not require any such command.

- Choose option 6, printer codes.
- Choose option 1, characters per inch, type the number then enter your proper printer control code for the elected character sizes. Repeat these steps for all commands which you intend to use (refer to Figure II for a list of possible commands).
- Choose option 2, lines per inch, and enter printer control codes for both available options. For six lines per inch enter **ESC 2** and for eight lines per inch enter **ESC 0**

```
File: AU.EP.FIG.1235          REVIEW/ADD/CHANGE           Escape: Main Menu
=====|====|====|====|====|====|====|====|====|====|====|====|====|====|====|===
NEW FUNCTION     #   STANDARD FUNCTION      PRESS        CONTROL CODES

UNI 1            1   Boldface begin         OA-B         ESC
                 2   Boldface end           OA-B         ESC ! CTRL-@

UNI 2            3   Underline begin        OA-L         ESC
or CONDENSED     4   Underline end          OA-L         ESC ! CTRL-D

BACKSPACE        5   Subscript begin        OA-O -B      CTRL-H
                 6   Subscript end          OA-O -E      ESC ! CTRL-@

DOWNLOAD         7   Superscript begin      OA-O +B      ESC % CTRL-A CTRL-@
                 8   Superscript end        OA-O +E      ESC % CTRL-@ CTRL-@
...............................................................................

^M^GElite+BOLD^^  ^E^4Italics+EMPHASIZED^^  ^W1ENLARGED^  ^DOWNLOAD SET^

^x1NLQ PRINT^  ^xØ^  ^-1Underlined^  ^!$Condensed+ENLARGED^  ^p1Proportional^

^P^Condensed^P^ ^P^^SØSuperscipt I^^T^  ^S1Subscript II^^T^   Back <<^^^^----
-------------------------------------------------------------------------------
Type entry or use @ or  cmds             Boldface Begin             @-? for Help
```

Figure III

```
Elite+BOLD  Italics+EMPHASIZED  ENLARGED  DOWNLOAD SET

NLQ PRINT  Underlined  Condensed+ENLARGED  Proportional

Condensed  Superscipt I  Subscript II  Back <<--
```

Figure IV

- Choose option 3, boldface, subscript and superscript, and enter the keystrokes under the heading Control codes in Figure III.
- Choose option 4, underlining, and again type in the keystrokes from Figure III.
- Press Escape (five times) to return to the main menu.

Always press ^ after the whole printer control code has been entered: Only by proceeding this way can you leave the menu. Don't press Return – it will be displayed as code.

If you make a mistake while entering a printer control code, press ^ and choose the same option again. Then answer No to the question "Is this OK?" and enter the correct keystrokes. Press ^ after the control code has been entered.

This is almost all that is necessary to install the printer control codes for all print sizes, type styles, typefaces and the other uses mentioned above.

Now you have to deal with two pairs of Universal codes, UNI 1 and UNI 2 (#1,2 and 3,4) for the standard functions Bold Begin/End and Underline Begin/End.

And with another two pairs of special codes (#5,6 and 7,8) for the standard functions Subscript Begin/End and Superscript Begin/End.

Remember that if the cursor is on ^ the name of the standard function always appears at the bottom of the screen, and this cannot be changed to the new ones. But after working a while with the new configuration you will find that this presents no problem.

The second new Universal code (UNI 2) is so constructed that in addition to its universal function it will act as printer control code for the condensed font.

Both codes are fine for embedding in your text practically all printer control codes, with the exception of those for NLQ font and backspace – you can't write control codes in the AppleWorks document.

The universal codes are described in Figure III under the names UNI 1 and UNI 2, and both have the same Escape code as the Begin function: This is not a mistake but a necessity. Their begin functions are really the same, but their End functions are differ- ▷

```
1  3                    4  2        1  3                          4  2
^M  ^G  Elite+BOLD      ^  ^        ^E  ^4  Italics+EMPHASIZED     ^  ^

1                  2                7                      8
^W1  ENLARGED      ^                ^  DOWNLOAD SET        ^

1                  2  1  2          1                      2
^x1  NLQ PRINT     ^  ^x0^          ^-1  Underlined        ^

1                        2          1                      2
^!$  Condensed+ENLARGED  ^          ^p1  Proportional      ^

3 4                1 2              3 4  1                      2  1 2
^P^  Condensed     ^P^              ^P^  ^S0  Superscript I     ^  ^T^

1                  2 1 2            5 5 5 5
^S1  Subscript II  ^ ^T^            Back  <<  ^ ^ ^ ^ ----
```

Figure V: Figure IV in detail

ent and you will have to call them differently – UNI with Control+B, UNI 2 with Control+L.

AppleWorks has these codes in a kind of stack – always enter the Begin command first. For some printer functions you have to put two printer control codes, both starting with ESC, before the characters.

For instance, if you need to print in the italic+emphasised typestyle, look again at the lower part of Figure III for examples of what to type, and at Figure IV for the printed result.

In the first example you will see:

^M^GElite+BOLD^ ^

with the position of the cursor marked with an arrow. And at the bottom of the screen in Figure III you can see the name of the standard function Boldface Begin as AppleWorks reads ^ under the cursor.

Note that in the following examples the spaces between the codes are only to aid understanding. When writing a real document, never put spaces within the code: The printer's firmware would translate them as part of the printing code and results would be unpredictable. However, you can put any number of spaces after the code, or between normal characters – as usual.

The next part of the Escape code sequence would be part of your document, and immediately after ^ you must type the necessary characters. To get the result shown in Figure IV, for example, you should do the following:

Press **Control+B** then type **M**; Press **Control+L** then type **G**; Type **Elite+BOLD**; and finally press **Control+L** and **Control+B** again.

This sequence will actually be sent to the printer as:

ESC M
ESC G
Elite+BOLD
ESC !
CTRL-@
ESC !
CTRL-@

You can't use Control+B M followed by Control+B G, because AppleWorks would translate the second Control+B command as being the end of the first one. Thus you have to use both UNI 1 and UNI 2 commands.

A further function of UNI 2 is to send the printer control code for condensed type. For example:

Control+L P Control+L Condensed
Control+B P Control+B

Here it is necessary to type **P** after pressing Control+L or Control+B for the first time: You have to distinguish somehow between the Begin and End functions of the same code. You can't simply use **Control+L Control+L** – or rather you can, but instead of printing the next piece of text properly in condensed mode, the printer will produce !, and any subsequent text in the current size.

Escape+P will send the command for printing in Pica mode, and the UNI 2 second code, Control+L (End), will send the next control command for printing in condensed style.

In UNI 1, the first command is the same as in UNI 2, and the second code, Control+B (End) will send the command for printing the Pica mode in normal size.

You must always use the second universal code (invoked by pressing Control+L) inside the first universal code (invoked by pressing Control+B) as in the above example, and never the other way round, for the proper execution of a printing command.

The printer control code used as Control+B End function (ESC ! CTRL-@) in the UNI 1 command (see Figure III) cancels all other printing codes with the following two exceptions:

- The command to print characters in subscript and superscript modes must always be cancelled with ESC T. You have to add this command after the UNI 1 End command, as in the examples in Figure III.
- Use of the Download font must be cancelled with the **ESC ! CTRL-@ CTRL-@** printer control command which is automatically embedded in the text with the END command for Download font printer control commands.

The example in Figure IV is more fully described in Figure V – you can see exactly what codes in what sequences were used to get the printed result. The numbers above the first printer control code refer to the new function number, as shown in Figure III.

However, these new codes have limitations enforced by AppleWorks itself.

All codes which you substitute for Subscript Begin or Superscript Begin (#5 and

```
File: AU.EP.ZK1                REVIEW/ADD/CHANGE              Escape: Main Menu
=====¦====¦====¦====¦====¦====¦====¦====¦====¦====¦====¦====¦====¦====¦====¦===
--------Chars per Inch: 13 chars
--------Right Margin:  0.0 inches
--------Left Margin:  0.8 inches
 This is the first line in NLQ mode. Max. 10 chars. per inch.
--------Chars per Inch: 10 chars
--------Left Margin:  1.0 inches
   Without the command '10 characters per inch' after first line, the
   AppleWorks would think, that you chose really 13 characters per inch,
   will format the document on the screen in accordance with it and will
   try to send this quantity to your printer. The result will be a funny
   printout. (Oh yes, not so funny as ugly.)
--------Chars per Inch: 14 chars
--------Left Margin:  0.7 inches
 And this is the first line in Draft mode. Max. 10 chars. per inch.
--------Chars per Inch: 10 chars
--------Left Margin:  1.0 inches
   The first line in the Draft mode begins with one space (now your
   command is for one character per inch more than for the first line in
   NLQ mode) and again with a maximum of 10 characters per inch. For
   different LM and RM values adapt also other values accordingly.
--------------------------------------------------------------------------------
Type entry or use @ or  cmds          Line 14  Column  2       @-? for Help
```

Figure VI

```
This is the first line in NLQ mode. Max. 10 chars. per inch.
Without the command '10 characters per inch' after first line, the
AppleWorks would think, that you chose really 13 characters per inch,
will format the document on the screen in accordance with it and will
try to send this quantity to your printer. The result will be a funny
printout. (Oh yes, not so funny as ugly.)
And this is the first line in Draft mode. Max. 10 chars. per inch.
The first line in the Draft mode begins with one space (now your
command is for one character per inch more than for the first line in
NLQ mode) and again with a maximum of 10 characters per inch. For
different LM and RM values adapt also other values accordingly.
```

#7) will be in force for all succeeding characters as they print, until either you stop their influence by the codes typed under the options Subscript or Superscript End (#6 and #8), or the current line ends.

Similarily, codes which you substitute for Boldface Begin or Underline Begin (#1 and #3) remain in force until stopped by codes typed in under options Boldface/Underline End (#2 and #4), or by the end of a paragraph. If you need more paragraphs in the same mode, simply repeat the code at the beginning of each paragraph.

Remember that in spite of this automatic ending of the printing codes, you must always give the second (End) command for codes #6 or #8 on the same line as the Begin command #5 or #7, and for codes #2 or #4 in the same paragraph as the Begin command #1 or #3. Otherwise, your printer may behave very unpredictably.

If you want to print a document in download font (or the NLQ set) you'll find AppleWorks less than cooperative.

Finally though, I found a solution, which is useful for other commands as well. Despite the warning in the AppleWorks manual not to enter codes that have a different meaning under the characters per inch option, it is possible to see on screen what your text will look like when printed.

```
File: AU.EP.ZK2                REVIEW/ADD/CHANGE              Escape: Main Menu
=====|====|====|====|====|====|====|====|====|====|====|====|====|====|====|===
--------Chars per Inch: 23 chars
--------Right Margin:  Ø.Ø inches
--------Left Margin:  Ø.4 inches
 This is the first printable line of your document which will be printed
--------Chars per Inch: 1Ø chars
--------Left Margin:  1.Ø inches
      without remarks after the printer control code DC1 and DC3 (disguised
      as '24 (23) chars per inch' commands) are sent.
--------Chars per Inch: 24 chars
      >These two lines with your unprintable remarks are on the screen only
      >and you can have an arbitrary quantity of such remarks.
--------Chars per Inch: 23 chars
--------Left Margin:  Ø.4 inches
 This is the first line of the next printed part of your document
--------Chars per Inch: 1Ø chars
--------Left Margin:  1.Ø inches
      Please note the arrangements for first lines (^if basic LM^^<^>^^Ø^):
      begin with one space, max.1Ø chars/inch, LM = Ø.4; for other LM values
      adapt these arrangements. PL naturally depends on all lines incl. non-
      printable ones, therefore adjust it accordingly (use OA-K for control)
-------------------------------------------------------------------------------
Type entry or use @ or  cmds          Line 4  Column 74        @-? for Help
```

Figure VII

```
This is the first printable line of your document which will be printed
without remarks after the printer control code DC1 and DC3 (disguised
as'24 (23) chars per inch' commands) are sent.
This is the first line of the next printed part of your document
Please note the arrangements for first lines ( if basic LM  < >  Ø ):
begin with one space, max.1Ø chars/inch, LM = Ø.4; for other LM values
adapt these arrangements. PL naturally depends on all lines incl. non-
printable ones, therefore adjust it accordingly (use OA-K for control)
```

Many possibilities

The left margin will jump aside a little (on display only) after you embed these printer control commands in a document, but this is a small price to pay for all the possibilities obtainable.

All the necessary printer control commands are in Figure II to be entered as described in the instructions for creating the custom printer driver under the heading Option 1 – Characters per inch.

Choose those control codes for character sizes not available on the Epson FX-85. I did this for commands to print with the Download font, the NLQ set or one of Epson's international character sets.

I intentionally used those options where the number of characters is greater than the smallest character size I can use, but at the same time with the smallest difference between numbers (for Begin and End) of characters per inch. These conditions are useful in overcoming one catch.

For example, with NLQ mode, AppleWorks will send the above codes to your printer, but also thinks that you want 13 characters per inch. Thus it will try to send the printer 104 characters within a single line if you have both margins set to one inch.

Because your NLQ characters would be in Pica size – 10 per inch – the printer will produce some very strange results as it tries to handle two mutually exclusive commands.

One solution is shown in Figure VI. If you send, via the Open Apple+O command for printing 13 characters on a line, **OA-O CI RETURN 13** (actually a command to enable printing with NLQ font) you have to cancel it immediately with another establishing, for example, 10 characters per inch.

This second command is necessary for correct printing. But in AppleWorks these two commands can't be together – there must be at least one real character between them. Note that the example in Figure VI is valid only for the left margins shown.

And the best idea? The simplest one is to divide two lines with different characters per inch with one character only, for example, with a comma. After printing, this can easily be covered with correction fluid.

The inability to put non-printing comments in a document can be resolved by using the proper printer control codes, disguised as the 23 and 24 characters per inch option.

Note though that to use these commands you'll have to set the Epson's DIP switch 2-1 to off. Then, when your printer is switched on, it will automatically be in the deselected state and will not print until **CTRL-1 (DC1)** is sent to it.

With AppleWorks this is no problem – simply enter **CHRS-Q** as the first printer control command under option 5, the Interface card.

But consider carefully whether your other software can also send control codes to the printer. Some popular software can't – Multiscribe and Printshop, for example.

To use the Select/Deselect function, see Figure VII. Again you can use the simple, small character solution described earlier. I have so many programs that can't send control codes that in the end I didn't use this option, mainly because of the DIP switch problem. If anyone knows a better way to include non-printing comments, let us all know!

And that's all. A long explanation, but don't think that these codes are difficult to use – after three or four documents I was working with them without any problems.

No problems

And don't hesitate to adapt the codes to your own needs. For example, if you often use underline and wish to reserve the Control+L command for it, perhaps use the backspace instead. Use the UNI 2 function (the standard subscript function) and enter the appropriate codes for the underline function to act normally.

Or you can add two more codes (4, 7 and 11 characters per inch are not used in Figure II, for example) to switch the paper-end detector on and off.

As a matter of fact, the printer control code #6 (Backspace End) in Figure III is quite unnecessary – I added it only in case code #6 was wrongly entered. Of course, you can use this location for other codes.

For your convenience, Figure VIII summarises characters you need to type after Control+B or Control+L for the appropriate printer control commands.

```
        After first <CTRL B> or <CTRL L> type following character(s)
                          for printer control codes:

 M    Elite          P    Pica           E    Emphasized     G    Bold
 4    Italics        W1   Enlarged       x1   NLQ+           xØ   NLQ-
 -1   Underlined     SØ   Superscript+   S1   Subscript+     T    Sub & Sup-
 1    7/72" spacing  p1   Proportional   !$   Condensed & Enlarged
```

Figure VIII

LEAP AHEAD OF THE GAME

The Gazelle

is an integrated communications program written in Assembly Language for APPLE IIe, IIc and IIGS

The Gazelle

offers BRILLIANT COLOUR – and a Viewdata Frame Editor with Carousel. Download as many frames as you like, edit them then carousel the display with timer control.

The Gazelle

is State-of-the-Art Prodos communications software

- EASY TO USE
  Select Commands with Mouse, Alphabetic or Cursor Keys
- UNIQUE FRIENDLY INTERFACE
  Pull-Down Menus
- WRITTEN BY EWEN WANNOP
  Well-Known brainy boffin
- DIRECT ACCESS TO HELP SCREENS
  from any point in program
- SENDS APPLEWORKS files and DOWNLOADS Email as AWP files
- STATUS BAR DISPLAY
  Shows Buffer, Speed, Date & Time
  Stays active while online
- FAST IN USE
  Memory-resident Program
- FILE CONVERSION from DOS 3.3
- CONTINUOUS SPOOLING
  Up to 30 Megabytes
- BUILT-IN CLOCK
  (for Mousecard & IIc)
- LARGE 44k CAPTURE BUFFER
- TEXT EDITING & MACRO FACILITIES

Leap sure footed with the GAZELLE

into File Transfers, Telecom Gold, Telex and Prestel!

GAZELLE

Special Trade-In Offer

Upgrade your existing comms
Send us the **original disc** from ANY comms package for a **£20 Trade-In!!**

Order by Cheque or Postal Order
£75.00 + VAT (£55 + VAT if with Trade-in)

From **KOLOUR SOFTWARE LIMITED**
52 The Spring. Market Lavington. Devizes. SN10 4EB
Telephone 0380 818667

The WIMP link

Apple is moving the IIgs and Macintosh families closer together. Here Cliff McKnight looks at the common user interface, the WIMP environment, which the two machines share.

I HAVE to admit that ever since I read about the Xerox Alto and saw the Xerox Star, ever since I laid hands on a Lisa, I've been a committed WIMP fan. Those Windows and Icons, that Mouse, those Pull-down menus turned my head from my first love, the Apple II. Now, with the advent of the IIgs, my affairs are reconciled.

I thought things were looking up when the mouse appeared for the IIc, but apart from MousePaint none of the software seemed to support it. I also flirted briefly with the Atari ST, but now I can see why they're trying to make that machine emulate a Mac.

From the point of view of a disciple, then, I'd like to sound a cautionary note about WIMPs in general, and icons and mice in particular.

Those of you who heard Alan Kay talk at the AppleWorld show last year will no doubt remember him recounting how at Xerox they soon realised the disadvantage of the light pen as a pointing device – after 20 seconds all the blood has drained from your hand. In comparison, the mouse was much easier to use. Moving it around the desk was easy, and when you stopped moving it the pointer stayed still.

The problems begin to appear when you consider not just the user and the tool, but the task and the environment in which the user, task and tool are interacting. I came across an interesting example of the environment playing a critical role when I talked to someone who was involved with Human Factors research in the navy.

It seems that mice are frowned upon in the naval context – not because sailors are too macho to use them but because the ship or submarine is likely to be tossing about quite severely. Under these conditions, devices like mice and light pens that are not screwed to the table will fall all over the place.

There are also studies which show that, under certain conditions, cursor keys produce "better" results. For example, John Ewing and others at Maryland University found that experimental subjects performed faster with arrow jump keys than with a mouse, and also preferred the keys. If you count the number of errors made, though, the mouse was superior.

In places like DisneyWorld, where the number of users is astronomical, neither mice nor light pens nor even keyboards stand a chance. Just about the only input mechanism that survives in those conditions is the touch screen. Think about it – no moving parts, no loose wires, nothing to go wrong as long as the screen is cleaned regularly to get rid of all the sticky fingerprints.

Although they're a comparatively recent innovation in computing, icons and the idea of representing things pictorially have been around for centuries.

As the philosopher and mathematician Gottfried Wilhelm Leibniz (1646-1716) observed: "In signs one sees an advantage for discovery that is greatest when they express the exact nature of a thing briefly and, as it were, picture it; then, indeed, the labour of thought is wonderfully diminished."

A good icon will produce the same reaction in most people, irrespective of their background, education or nationality.

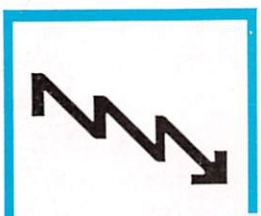


Figure I: The Ready,Set,Go 3 icon

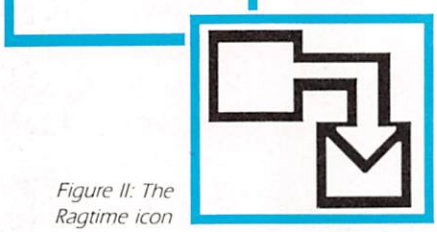


Figure II: The Ragtime icon

Unfortunately, the number of "good" icons appears to be limited. For example, what does this icon represent?

Obviously it represents the wrath of God – users selecting this option will be subjected to a lightning strike. In reality, it's the icon used in Ready,Set,Go 3 to get text to flow from one column to another. The same function is indicated in Ragtime by:

Pagemaker has a different way of performing the same function, while MacAuthor simply requires you to select "Allow continuation frames" from one of the set-up menus and will then flow the text automatically.

If the RSG3 icon had been perfect, its makers would no doubt have sued other companies who used it so it's not surprising that different packages use different icons. Also, since MacAuthor can do the job automatically and without icons, it's not clear that any icon is necessary in this case.

Even with the right environment and the right task, implementing a WIMPs interface doesn't guarantee a smooth ride. The Gem operating system is a case in point. I use an IBM PC/AT regularly and sometimes run various software under Gem. While I have to agree that I prefer the Gem environment to MSdos (there are times when I'd prefer a hole in the head to MSdos, but that's a different story), there is a world of difference between, say, Gem-Paint and MacPaint.

The difference comes from the fact that the Mac and the IIgs have all the toolbox and quickdraw routines built into the firmware. The poor old PC has to try to produce the same effect with little other than software. Digital Research have done a grand job in many ways, but I can't help feeling it's like trying to race a double-decker bus around Brands Hatch.

Wimps aren't the panacea which many software houses would have you think them to be. At the end of the day, naff software is naff no matter how many windows you can have open simultaneously, no matter how many buttons there are on your mouse.

One of the early spreadsheets I saw on the Atari ST required you to select a cell with the mouse, enter the data from the keyboard, then finish the entry with the mouse. Entering a lot of data was tedious in the extreme, constantly moving from mouse to keyboard and back. In these conditions, even in a WIMP environment it's important to have keyboard equivalents. Even the old stalwart VisiCalc on the Apple II was preferable to the ST package.

The general point is that it is not sufficient to consider just the interface between the user and the tool. System designers must also consider the task which the user is performing and the environment in which the user-tool-task interaction is taking place.

Next time you hear someone knocking the WIMP environment, listen carefully – the chances are they are telling you about system designers who didn't do their job properly rather than just knocking machines like the IIgs and the Mac.

If you have something to say which directly concerns both machines, why not say it through the pages of Apple User? It may be about common problems, it may be about an application where the two machines work together, it may be about a software package available for both machines. As long as what you say is relevant to both groups of users, the soap-box is yours for a month.

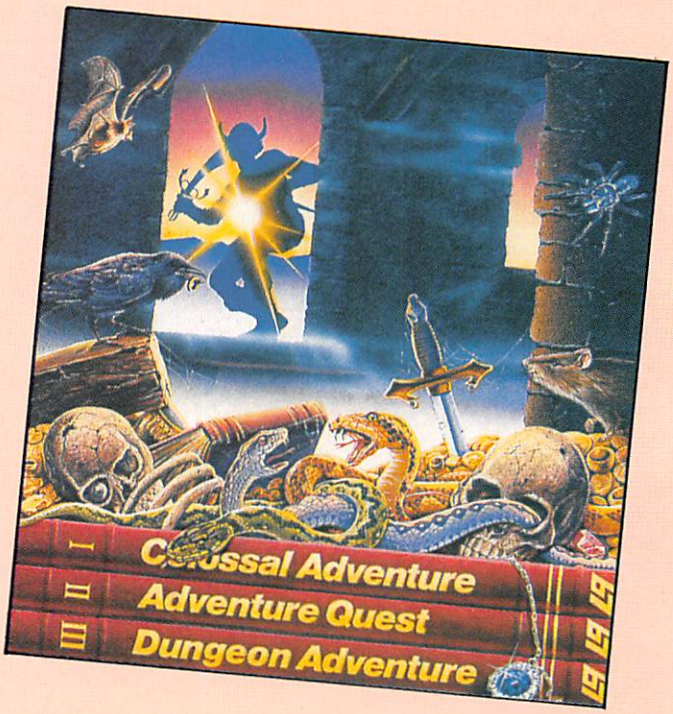
AppleUser SPECIAL OFFERS!

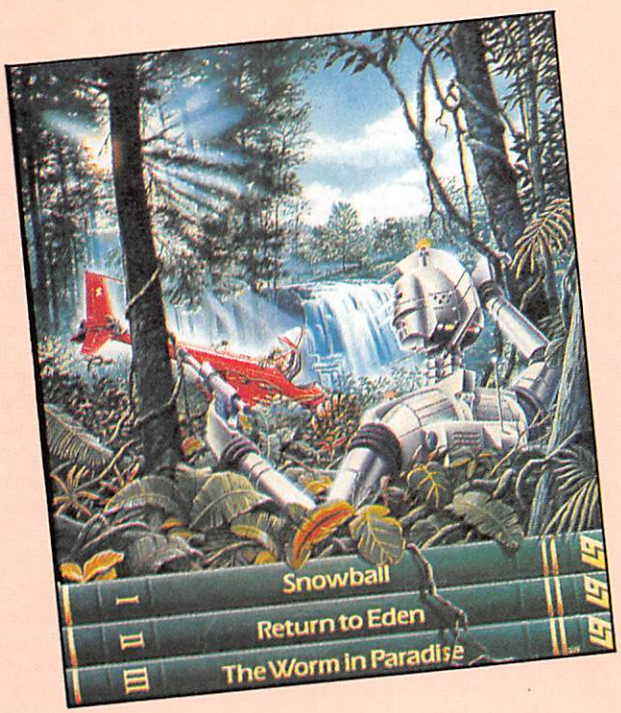
Two top adventure trilogies for you to play

Award-winning software house Level 9 has extensively re-written some of their best-selling adventures, and released them in two trilogies: Jewels of Darkness and Silicon Dreams.

In the Jewels of Darkness trilogy you start with Colossal Adventure, containing all the treasures, creatures, rooms and puzzles of the mainframe original.

In Adventure Quest you must discover the Old Roads to the Dark Tower, Fortress of the Demon Lord. Only there can you defeat him. There's magic in the air in Dungeon Adventure. Can you discover the treasure while facing the perils of skeletons, carniverous jellies and orcs?





The first adventure in the Silicon Dreams trilogy is Snowball. You awake from suspended animation to find your spaceship on a collision course with Eden. In Return to Eden you must prevent the defence robots from destroying your ship. You have lost your memory in the Worm of Paradise, and you may have to join the governing party to regain it.

Each features:
Over 600 illustrations
New language interpreter
Huge 1,000 word vocabulary
Multi-command sentences
Ultra fast response times
64 page novel and 12 page guide

Apple IIe or Mac	RRP	Special Reader Offer	You Save	Offer Inc. Subscription	You Save
Silicon Dreams	£19.95	£14.95	£5	£30.95	£7
Jewels of Darkness	£19.95	£14.95	£5	£30.95	£7
Silicon Dreams and Jewels of Darkness	£39.90	£27.90	£12	£41.90	£16

You save £5 when you buy one of these packs or £16 if you buy both

TO ORDER PLEASE USE THE FORM ON PAGE 73

Finding your way round the Mac SE

Cliff McKnight gives his first impressions of Apple's new machine

LET'S get one thing clear right from the start – this is neither a review of the Mac SE, nor is it a benchtest report. It is some first impressions after using an SE for a couple of weeks.

The machine in question was kindly loaned by Apple UK. Unfortunately it arrived just before I went to the States for a fortnight, and it was only available for short-term loan. According to Apple, there was a queue of people waiting to borrow it, but I'm not sure if this suggests a shortage of machines or simply a small pool of loan machines. I'm certain there are lots of people who'd like to borrow one.

No matter, because during my absence the machine was heavily used by the rest of the family who normally fight with me for access to our 512k Mac. By the time I returned, the 20Mb internal hard drive was bulging at the seams.

The machine arrived without any manuals or startup discs so I was forced to use the latest issue of the MacPlus system and finder rather than the new versions which are supposed to accompany the SE. Fortunately I also had a hard drive installation disc, otherwise I might not have been able to use the internal 20Mb drive.

The lack of manuals meant that I never did discover what the large key on the top of the keyboard was supposed to do. I've read elsewhere that it is a switch, but it certainly didn't switch anything that I could discern.

I tried it in combination with every other key and with several multi-key combinations but to no avail. I thought of dismantling the keyboard to see if it was even connected, but thought this might not be considered appropriate use of a loan machine.

The lack of the correct system file meant that desk accessories like KeyCaps showed the wrong positions for lots of the keys. Presumably the keyboard mapping is correctly represented in the new system. Although the £ was shown above the 3 on the keyboard, pressing Shift+3 printed a #, and Option+3 was required to produce £.

While I'm on about the keyboard, I might as well say that it has quite a nice feel to it, being lower than the "normal" Mac keyboard. However, the spacebar is a disgusting little thing which feels like it's sat on chewing gum rather than springs. It certainly didn't seem to travel as far as the other keys and I kept wondering if it had actually worked.

The keyboard is obviously geared to the Big Blue World, as evidenced by the addition of such keys as Escape and Control in addition to the more usual Command and Option. No doubt the next stage will be to relabel Return as Enter and add NumLock.

One thing I do like is the new mouse. The phrase which springs to mind is "adverse camber" because it seems to be sloping the "wrong" way. In fact, I found it a lot more comfortable than the old Mac mouse. It's less bulky, more dainty, just right for my little hands. The mouse can be plugged into one of the two ports on the back of the machine or it can be daisy-chained from a port on the keyboard.

Once I'd set up the hard drive it worked very well. It powers up and accesses much faster (subjectively at least) than even an external SCSI drive, and orders of magnitude faster than an HD20. Towards the end, though, it was occasionally producing some strange high-frequency, metallic, vibrating sounds while idling.

Not knowing what the drive had been subjected to earlier in the queue, it's possible that the odd noises were the result of some former rough treatment or the amount of travelling it had done. However, for a new-looking machine (which, after all, can't be very old) the noises were a bit worrying, even though they didn't seem to affect performance.

While on the subject of noise, I have to say that the internal fan was an infernal nuisance. While such a noise might not be too intrusive in a normal office environment, late at night in a quiet house it sounded very loud – a bit too loud to have in the same room as the television.

I also had a slight problem with the internal 800k drive. It suddenly refused to read a particular 400k disc which it had previously read perfectly and which neither a 512/400k nor MacPlus 800k drive had any problems with.

Another problem I'd never previously encountered occurred with a folder on the hard drive. It suddenly wouldn't open and bombed the system every time I tried to open it. I used the DeskTop utility to move all the files out, whereupon the folder would then open, but the replacement wouldn't.

After half an hour of trial and error I managed to reconstitute the folder and files under a different name, threw away the original and renamed the new folder back to the old name. It's never happened to me before on various Macs with various hard drives, but it's the kind of problem I'd attribute to the system or finder. Maybe if I'd had the right files . . .

Lacking the startup box of manual and discs, the machine also lacked the interrupt/reset switch which gets attached to the side. Without it, I had to cure lockups by powering down – a bit drastic, and also the lack prevented me using the Crash recovery utility.

I didn't get to plug anything into the System Expansion slot which gives the machine its monogram, but I did receive information about the HyperCharger 020 from General Computer Corporation. This is a 16MHz 68020-based processor and ram expansion board which claims to quadruple the speed and performance of the SE.

There is an optional 68881 floating point co-processor, and the 1Mb on-board ram can be expanded to 4Mb. The price of the basic board is expected to be a mere £1,490, but at the time of writing I didn't have prices for the other options. If you're interested, P&P should have the boards in stock by the time you read this.

In fact the slot is the only real difference between the Mac Plus and the SE as far as I'm concerned. I enjoyed the luxury of the 20Mb internal hard drive, but for the price difference I'd settle for an external SCSI drive and a Mac Plus.

I have all the IBM compatibility I need – a wire down which I squirt text files from one machine to the other. However, if you need real compatibility, the SE is certainly an easier route than MacCharlie.

If Apple is looking for a good home for the loan machine once it's done the rounds I'd be more than happy to oblige. I'd even offer to swap it for my 512k machine, maybe. However, I'm not rushing out to replace my "work" MacPluses with SEs just yet. □

Apple can monitor the airwaves

J.J. Taylor's assembler program will turn your Apple into a radio teletype receiver

RADIO teletype was one of the earliest methods of data communication over the airwaves, and indeed is still in common use today.

It involves the use of the mechanical teletype machines very familiar in all Post Office and Government offices. A message is typed character by character on a keyboard, encoded and sent by radio signal to a receiving teleprinter which decodes it and prints it out.

Nowadays these mechanical teleprinters and rolls of paper are being replaced by micros and VDUs. And the Apple is quite capable of decoding these messages with the appropriate receiving equipment.

Such messages are to be found all over the radio spectrum, and have a distinctive "tinkling" sound which is unmistakeable once heard. The transmissions originate from many sources, including the press agencies, embassies, radio amateurs, weather stations and naval and military traffic.

Ignoring the transmission of these messages, which is beyond the scope of this article, and indeed only of interest to the licensed radio amateur, I will explain the simple assembler program I use to receive and print RTTY messages with the Apple II.

Characters represented by a serial stream of binary 1 or binary 0 bits are sent as high (known as Mark), and low (Space) audio frequencies. When received at the speaker of the radio receiver, they are sent via a decoder, which converts these frequencies back to Mark and Space signals (as DC voltages), recognisable by the computer.

The signals are sent by a radio carrier wave which alternates between the two frequencies. The spacing between them is usually between 170 and 850Hz, 425Hz being favoured by press agencies. The signal must be received as single sideband, so a a good quality receiver capable of resolving SSB is essential.

In standard RTTY each character is represented by an agreed international code of five data bits, to which is added a Start and Stop bit. This, using one character as a figures-shift and one as a letters-shift gives a functional alphanumeric character set plus a few control characters. The letters-shift, for example, is represented by the five bit code 11111 or decimal 31 and figures-shift by 11011 or decimal 27 in the Baudot code. Letter Y or figure 6 is represented by 10101, depending on whether letter or figure-shift is in force.

This standard RTTY, known as Baudot or Murray code, is commonly sent at a speed of 45.45 or 50 baud (60 or 67 wpm), though other baud rates may be seen.

Depending on the baud rate at which a character is transmitted, each will be composed of a Start unit, five time units and a Stop unit. The Start pulse is always the same length as the Data pulses, while the Stop pulse is usually 1.5 times the Start pulse. For a 50 baud transmission the Start and Data pulses are 20 milliseconds in duration, and the Stop Pulse 30 m/s. For 45 Baud, corresponding times are 22 and 33 m/s.

To decode, the program will wait until the signal goes to Space which will be commencement of the Start pulse, and pause half a time unit, when we will be in the middle of the Start pulse. Now sample after each of five successive time units, pushing the logic state into a character store. Now the program pauses for a final 1.5 pulse units, leaving us with half a unit before the end of the Stop pulse.

This half unit gives us ample time to process our received character, and place it on-screen before looking for the beginning of the next Space pulse. The serial bit stream obtained from the five logic states is rotated left in the usual manner to give the Baudot byte in the character store. And, from this, the appropriate Ascii character is obtained from a look-up table, and placed to the screen.

The timing for the units is obtained from the Apple Monitor WAIT routine located at $FCAB. This is a fairly well documented routine in which the delay in microseconds is:

$$1/2(26+27A+5A^2)$$

where A is contents of Accumulator.

From this it is relatively easy to ascertain that for 45 baud a 22 m/s delay needs 91 to be placed in the accumulator, and for 50 baud a 20 m/s delay 86, before the CALL to WAIT.

The output from the decoder is directed at TTL levels to a one bit input on the game paddle..SWO (Pin 2) of the game paddle, and the program samples the logic at this switch. The decoder, of which there are varied designs, may be constructed (if you have the ability), or bought ready-built. I use the CP-1 modem from AEA, who have several models including custom cabling for plugging into your internal game paddle socket.

The assembler program is fairly well annotated and should be easy to follow. The program, which is assembled at $4000, allows for change of baud rate between 45 and 50, and a Clear Screen command. Included in the top left of the screen is a simple tuning device, two asterisks which will flash when a RTTY signal is tuned in: The characters will then appear.

The binary program should be BSAVEd and BLOADed at $4000 and the basic frame program run after installation.

I use a good quality single side band communications receiver on upper side band, and particularly enjoy the press agency emissions which are frequently the basis of the next days international news. Many are in English but French, German, Spanish, and Italian broadcasts may appeal to language buffs.

Finally, I am not a professional programmer, so no doubt the program could be enhanced. However, it is functional and does what I set out to do, and I hope it may create another diversion for the Apple user.

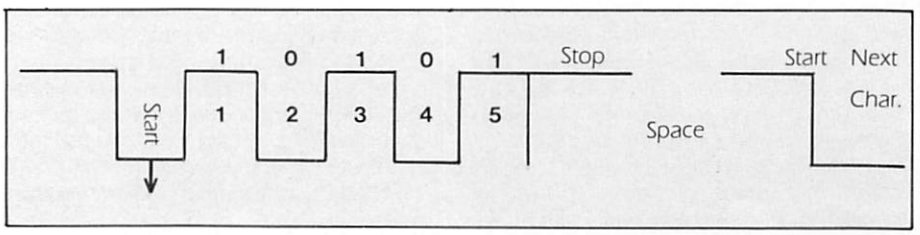


Figure 1: Example character Y

Listing I

```
:ASM

                1000        .OR $4000
                1010
                1020 ********************************
                1030 *                              *
                1040 *  BY J.J.TAYLOR   TEIGNMOUTH  *
                1050 *         VERSION 2.0          *
                1060 *                              *
                1070 *    RADIO TELETYPE RECEIVE    *
                1080 *     PROGRAM FOR APPLE II     *
                1090 *                              *
                1100 *                              *
                1110 * '5' CHANGE BAUDRATE TO 50     *
                1120 * '4' CHANGE BAUDRATE TO 45     *
                1130 * 'C' CLEAR TEXT               *
                1140 *                              *
                1150 * KEY SPACE BAR EXIT TO BASIC   *
                1160 *                              *
                1170 ********************************
                1180
FC58-           1190 CLEAR  .EQ $FC58
0006-           1200 STORE  .EQ $6        ;CHARACTER
001A-           1210 SHIFT  .EQ $1A       ;LTTRS/FIGS.
00CE-           1220 TEMP   .EQ $CE
001C-           1230 BAUD1  .EQ $1C       ;1/2TIME UNIT
001D-           1240 BAUD2  .EQ $1D       ;1 TIME UNIT
C061-           1250 SW0    .EQ $C061     ;SWITCH INPUT
FDED-           1260 COUT   .EQ $FDED
FCA8-           1270 WAIT   .EQ $FCA8
E003-           1280 BASIC  .EQ $E003
C000-           1290 KBD    .EQ $C000     ;READ KEY
C010-           1300 KBDSTB .EQ $C010     ;CLEAR STROBE
                1310
                1320 *-----------------------
4000- 20 58 FC  1330 BEGIN  JSR CLEAR
                1340 *-----------------------
4003- A9 00     1350 GO     LDA #0        ;START WITH
4005- 85 1A     1360        STA SHIFT     ;LTTRS SHIFT
4007- A9 40     1370        LDA #64       ;SELECT 45BD
4009- 85 1C     1380        STA BAUD1     ;TO
400B- A9 5B     1390        LDA #91       ;COMMENCE
400D- 85 1D     1400        STA BAUD2
                1410 *-----------------------------
400F- AD 00 C0  1420 BEG    LDA KBD       ;COMMANDS
4012- 2C 10 C0  1430        BIT KBDSTB
                1440 *-----------------------------
4015- C9 20     1450        CMP #$20      ;SPACE BAR ?
4017- D0 03     1460        BNE BD50      ;NO
4019- 4C 03 E0  1470        JMP BASIC     ;FINISH.
                1480 *-----------------------------
401C- C9 B5     1490 BD50   CMP #$B5      ;5 FOR 50BD?
401E- D0 15     1500        BNE BD45      ;NO
4020- A9 3D     1510        LDA #61       ;DELAYS FOR
4022- 85 1C     1520        STA BAUD1     ;
4024- A9 56     1530        LDA #86       ;(20MSECS)
4026- 85 1D     1540        STA BAUD2     ;50 BAUD
4028- A9 35     1550        LDA #$35      ;POKE 50BD
```

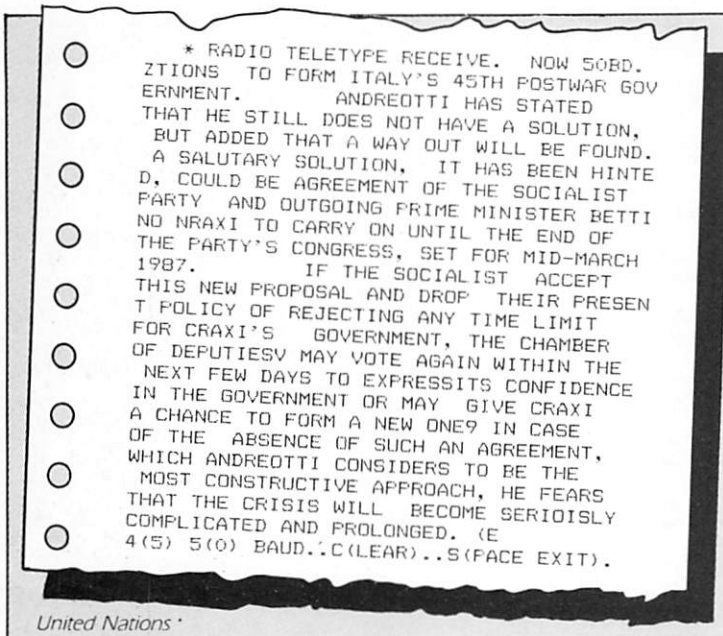
* RADIO TELETYPE RECEIVE. NOW 50BD.
Y O, JULY 21 ADN - NEARLY 500 JAPANESE
ORGANIZATIONS HAVE ALREADY SIGNED A
DECLARATION OF THE JAPAN COMMITTEE FOR
SOLIDARITY WITH THE PEOPLES OF ASIA,
AFRICA AND LATIN AMERICA PROTESTING
AGAINST THE APARTHEID SYSTEM IN SOUTH
AFRICA. THE DECLARATION CALLS
FOR THE IMEDIATE ABOLITION OF APARTHEID,
A HALT TO THE OPPRESSION AND MURDER
OF PEOPLE DEMANDING LIBERATION AND APPEA
LS TO THE BOTHA REGIME TO IMMEDIATELY
RELEASE ANC LEADER NELSON MANDELA AND
ALL OTHER POLITICAL PRISONERS. ZZ JA/JA
ADN 25 E 26 INTERNATIONAL
FORUM OF SCIENTIFIC WORKERS OPENDS
M O SC O W, JULY 21 ADN - AN INTERNATIO
NAL FORUM ''SCIENCE, TECHNOLOGY, PEACE''
BEGAN HERE ON MONDAY. ABOUT 500 PARTICI
PANTS FROM SIXTY COUNTRIES MEET FOR
THREE DAYS OCCASIONED BY THE FORTIETH FO
UNDATION ANNIVERSARY OFTHE WORLD FEDERAT
ION OF SCIENTIFIC WORKERS.
4(5) 5(0) BAUD..C(LEAR)..S(PACE EXIT).

East German News Agency

```
402A- 8D 22 04  1560        STA $422      ;TO
402D- A9 30     1570        LDA #$30      ;SCREEN
402F- 8D 23 04  1580        STA $423      ;ADDRESS
4032- 4C 0F 40  1590        JMP BEG
                1600 *-----------------------
4035- C9 B4     1610 BD45   CMP #$B4      ;4 FOR 45BD ?
4037- D0 15     1620        BNE CLR       ;CLR SCR ?
4039- A9 40     1630        LDA #64       ;DELAYS FOR
403B- 85 1C     1640        STA BAUD1
403D- A9 5B     1650        LDA #91       ;(22MSECS)
403F- 85 1D     1660        STA BAUD2     ;45 BAUD
4041- A9 34     1670        LDA #$34      ;POKE 45BD
4043- 8D 22 04  1680        STA $422      ;TO
4046- A9 35     1690        LDA #$35      ;SCREEN
4048- 8D 23 04  1700        STA $423      ;ADDRESS
404B- 4C 0F 40  1710        JMP BEG
                1720 *-----------------------
404E- C9 C3     1730 CLR    CMP #$C3      ;CLR SCREEN ?
4050- D0 03     1740        BNE NO
4052- 20 58 FC  1750        JSR CLEAR     ;CLEAR
                1760 *-----------------------
```

* RADIO TELETYPE RECEIVE. NOW 50BD.
WHWO JULY 21 KXINHUA) AA ROBBERY, MURDE
R AND KIDNAPPING WHICH TOOU PLACE
IN WEST BEIRUT IN THE PAST FEW
DAYSUHAVE DISTURBED THE SECURITY PLAN
IMPLEMENTED FOR TWO WEEJ UNDER THE
HELP OF SYRIAN MILITARY PERSONNEL.
GUNMEN CARRYING A MACHINE GUN
AND SOME SILENT PISTOLS ROBDED A BANK
IN WEST BEIRUT THIS MORNING AND ESCAPED
ON A CAR HAVING NO LICENS PLATE.
ON SATURDAY, THREE LEBANESE DOCTOR
S AND ON ADMINISTRATOR OF THE HOSPITAL
OF AMERICAN UNIVESITY IN WEST BEIRUT
WERE KILLED AND ANOTHER THREE WOUND
ED WHEN FOUR GUNMEN STOPPD THEIR
MOSPITAL BUS, CLIMBED ABOARD AND RAKED
THE 40 PASSENGERS WITH GUNFIRE.
TEAHERS AND STAFF OF AERICAN
UNIVERSITY STAGED A STRIKE TODAY TO
PROTEST AGAINST THE AMBUSH ON THE
BUS WHICH WAS CARRYING DOCTORS,
NURSES AND STAFFERS FROM WEST TO
4(5) 5(0) BAUD..C(LEAR)..S(PACE EXIT).

New China News Agency



United Nations'

```
4055- A9 00    1770 NO     LDA #0       ;ZERO
4057- 85 06    1780        STA STORE    ;CHAR.STORE
4059- A2 05    1790        LDX #5
405B- AD 61 C0 1800 FLAG   LDA SW0      ;SAMPLE SW0
405E- 30 FB    1810        BMI FLAG     ;TILL SPACE
4060- A5 1C    1820        LDA BAUD1    ;PAUSE 1/2
4062- 20 A8 FC 1830        JSR WAIT     ;TIME UNIT
4065- A5 1D    1840 TIMEU  LDA BAUD2    ;PAUSE 1
4067- 20 A8 FC 1850        JSR WAIT     ;TIME UNIT
406A- AD 61 C0 1860        LDA SW0      ;SAMPLE LOGIC
406D- 48       1870        PHA
406E- 30 0D    1880        BMI MARK
4070- A9 2A    1890        LDA #$2A     ;POKE * TO
4072- 8D 03 04 1900        STA $403     ;SCREEN
4075- A9 20    1910        LDA #$20     ;CLEAR *
4077- 8D 00 04 1920        STA $400     ;FROM SCREEN
407A- 4C 87 40 1930        JMP CON
407D- A9 2A    1940 MARK   LDA #$2A     ;FOR CRUDE
407F- 8D 00 04 1950        STA $400     ;TUNING
4082- A9 20    1960        LDA #$20
4084- 8D 03 04 1970        STA $403
4087- 68       1980 CON    PLA
4088- 0A       1990        ASL          ;BAUDOT CODE TO STORE
4089- 26 06    2000        ROL STORE    ;(A FIVE BIT CODE.)
408B- CA       2010        DEX
408C- D0 D7    2020        BNE TIMEU
408E- A5 1D    2030        LDA BAUD2    ;NOW
4090- 20 A8 FC 2040        JSR WAIT     ;STOP
4093- A5 1C    2050        LDA BAUD1    ;PULSE TOTAL
4095- 20 A8 FC 2060        JSR WAIT     ;1.5 UNITS
               2070 *-------------------------
4098- A5 06    2080 GO1    LDA STORE    ;PROCESS CHAR
409A- C9 1B    2090        CMP #27      ;FIGS SHIFT ?
409C- D0 07    2100        BNE LET      ;NO
409E- A9 20    2110        LDA #32      ;YES SO
40A0- 85 1A    2120        STA SHIFT    ;OFFSET.
40A2- 4C 0F 40 2130        JMP BEG      ;NEXT CHAR.
               2140 *-------------------------
40A5- A5 06    2150 LET    LDA STORE
40A7- C9 1F    2160        CMP #31      ;LTTRS SHIFT?
40A9- D0 07    2170        BNE SETX
40AB- A9 00    2180 LETS   LDA #0       ;YES NIL OFFSET
40AD- 85 1A    2190        STA SHIFT
40AF- 4C 0F 40 2200        JMP BEG      ;NEXT CHAR.
               2210 *-------------------------
40B2- 18       2220 SETX   CLC
40B3- A5 06    2230        LDA STORE    ;ADD STORE
40B5- 65 1A    2240        ADC SHIFT    ;TO SHIFT
40B7- 85 06    2250        STA STORE
40B9- AA       2260        TAX          ;TO X REG
               2270 *-------------------------
40BA- BD C5 40 2280 CARON  LDA LOOKUP,X ;CHAR.TABLE
40BD- 09 80    2290        ORA #$80     ;ASCII OUT
40BF- 20 ED FD 2300        JSR COUT
40C2- 4C 0F 40 2310        JMP BEG
               2320 *-------------------------
40C5- 00 54 0D
40C8- 4F 20 48
40CB- 4E 4D 0A
40CE- 4C 52 47
40D1- 49 50 43
40D4- 56 45 5A
40D7- 44 42 53
40DA- 59 46 58
40DD- 41 57 4A
40E0- 00 55 51
40E3- 4B 00 00
40E6- 35 0D 39
40E9- 20 23 2C
40EC- 2E 0A 29
40EF- 34 26 38
40F2- 30 3A 3B
40F5- 33 2B 24
40F8- 3F 27 36
40FB- 21 2F 2D
40FE- 32 07 00
4101- 37 31 28
4104- 00       2330 LOOKUP .HS 00540D4F20484E4D0A4C524
749504356455A4442535946584157 4A0055514B0000350D3920232
C2E0A2934263830 3A3B332B243F2736212F2D32070037312800
```

Listing II

```
5    REM BASIC Frame for RTT
     Y
10   HOME
20
   : INVERSE
   : PRINT "    Radio Telety
     pe Receive.  Baud ";
   : POKE 1058,52
   : POKE 1059,53
30   VTAB 24
   : HTAB 1
   : PRINT "4(5) 5(0) Baud..
     C(lear..S(pace  Exit).
     ";
   : NORMAL
   : POKE 34,1
   : POKE 32,23
   : HOME
40   CALL 16384
50   REM  J.J. Taylor
60   REM  Copyright 1987
```

Apple *User* Classifieds

Has upgrading your computer given you hardware you no longer need? Or have changing interests left you with unwanted software? Then THIS is the place to advertise your surplus items. *Apple User* readers are always on the lookout for a bargain and this is the first place they look!

An added bonus!

Your advert will also be automatically displayed on MicroLink, the electronic mail service operated in association with Telecom Gold. This means it will be seen by thousands of computer enthusiasts who can send an instant response.

Classified advertisements will be accepted under the following conditions:

- This service is EXCLUSIVELY for the use of private readers. No trade ads will be allowed.
- To avoid encouraging software piracy, all ads will be carefully vetted before they are accepted.
- Ads can only be accepted on this form (or a photocopy of it).
- There is no maximum to the number of words you include in your ad. If there is insufficient room on the form, continue on a separate sheet of paper.
- The cost is 20p per word, with a minimum of 10 words.
- We GUARANTEE your ad will appear in the July issue (on sale June 24) providing it is received by May 29.

● Titan Accelerator £100, Cis-Cobol Includes Forms £150. Both Apple Tel: 021 358 7572. Between 7.30 pm and 10.30 pm.

● Pinpoint, Pinpoint Spelling Checker. New Sealed. Free Ramkit On Registration £80 / both. Marlow (06284) 2817

● Giveaway Disks 100 SSDD 5¼" used disks in 10 plastic flip library cases £50. Also some software. Tel: Nigel 01-586 7285.

● Manuals or Copies wanted for Mac+ Super Draw McPaint Pagemaker Switcher McFile, McWord, McWorks, McDraw and Mcdeknife £5.- offered each Copy = Manual. Contact R Schuurman, P.O.Box 54011, 3008JA, Rotterdam, The Netherlands.

● Apple 11+ Green Monitor. 16K Language Card RGB Card Clock Card. 2 Apple Drives Versawriter Tablet Silentype Printer And Interface. Paddles, Discs, Manuals, £450 Telephone 0273 455232

● Back Copies Apple User 3 Yours A+ 2 Yours Offers. Tel: Nigel 01 586 7285

● Apple II+ £399, HAL 4-User 10MB Capricorn £595, Videx Enhancer II £59, SweetP Colour Plotter £129, Kaga 12'' Colour Monitor (Apple/IBM), £125, Apple II+ Colour Card £59, Aristocard Parallel Interface £29, Zappler Serial / Parallel Interface £39, CCS Synchronous Serial Card £49, Oki 82A Tractor £19. (+VAT). Tel: Windsor 855073.

● Apple IIe / Mac Compatible Symbiotic Hard Disk £350 Or Exchange Printer + Floppy. Tel: 01 769 0737 Evenings.

● Europlus Complete System For Sale Tel: Chesterfield 826302 For Details.

● Apple IIe 128K Complete System With Monitor, Disk Drives, NEC Printer, Joystick. The Receipts For This Lot Total Over £2200! Plus Manuals, Reference Books, Disks, Software And Five Years Issues Of Windfall / Apple User. Bargain £550. Tel: Allen Harris For Details On Bristol 0272 47865.

● As New PFS File And Report With Manual For IIe. Offers Or Swap For IIe Prodos Clock. Tel: 0705 452699.

● Two Rana Disk Drives Need Adjusting £35. Buyer Collects. Tel: 01 208 0809.

● Apple II+ Disk II, 640K Drive, Monitor, Printer, Z80 CP/M, RAM Expansion, Colour, 80 Cols, Wordprocessors, Languages, Games, Etc. As New, £475 ono. Tel 069260-493 (Norwich).

● WS3000 Modem, autodial, auto-answer, Hayes compatible V21,23 With Mac terminal, software manual and warranty. Almost new. Ring (0865) 723989

● Apple II Peripherals Discdrive £90. Controller £25. Epson MX80 + Grappler+ £145. Monitor £45. Grappler+ Communications £32 each. Buffered Grappler+, AD/DA 8 bit £60 each. Speechcard Z80 Numeric keypad £24 each. ALF 9 voice Musiccard Mockingboard Videx 80 Clockcard £45 each. Graphic Mouse £45. 128k Saturn RAM, IEEE-488 GPIB, IC tester £65. Software. Appleworks 1.3 £80. Peachtree Sales & Purchase £60. Format 80 Enhanced £40. All unused originals. Tel: 01 736 7809

● Apple Books: 33 Programs, 101 tricks, Creative Apple £6. Tel: 061 773 3965.

● Apple 2E 128k colour card twin disc drive monitor 3 £595. Apple 2, Apple monitor twin disc drive Seikosha GP-100A printer £395 both units excellent. Wycombe Cameras, 45 King Edward Court, Windsor, Berks. SL4 1TG

● Macintosh Plus for sale, 1MB memory 800k drive. Unexpired warranty, £1,400. Tel: 0865 723989.

					10 words £2.00
					15 words £3.00
					20 words £4.00
					25 words £5.00
					30 words £6.00

Cheque enclosed for £_____

Name_____ Address_____

POST TO: Apple Classifieds, Europa House, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.

A day in the life of a disc

Robert Neale and Colin Foster examine disc functions in Part Eight of their CP/M series

LET'S take a close look at what discs are, how they work and how CP/M stores information and programs on them. The basic Apple II disc drive uses 5.25in, single sided discs. This means that although we can use both sides of a disc our drive can only look at one side at a time without us turning the disc over.

A floppy disc is simply a disc of plastic, 5.25in in diameter, coated with the same sort of magnetic material as a cassette tape. But before the computer can use a disc it must be formatted. This process writes a preset data pattern all over the disc, dividing it up into areas which the computer can then address by number. These areas are called sectors and on the Apple 5.25in discs they can each hold 256 bytes of data, that is 0.25k.

On a record the music is engraved serially in one long line which spirals in from the outside of the record to the centre in order to pack as much information in as possible. We could obviously use a similar system to store our sectors on a magnetic disc.

However, unlike a record we rarely want to listen to the first half of the disc to get to some information stored in the middle. Also, without some sort of complicated indexing system the computer would not be able to go to a particular sector if they were recorded serially. Instead, a slightly different system is used. We split the disc up into 35 concentric circles of data called tracks, rather than a single long track.

Each track then holds 16 sectors of information and we can get at any one of them effectively as fast as any other by simply stepping the read head of the disc drive out to the required track and waiting till the sector we want passes under it, thus allowing us to read it. As the disc rotates at 300 rpm this does not take long.

This system is called random access to distinguish it from serial access as used in records and tapes. The first track, called track 0, is the outermost circle of the disc and track 34 the innermost. Different computers may use variations on this by having a different number of tracks.

40 and 80 tracks per side is common and different sizes of sectors, anything form 128 bytes to 1Kbyte, with between 4 and 26 sectors per track. This is one reason for incompatibility between discs from different computers. These days these differences are caused more by the whims of hardware manufacturers than by any inherent requirements of different operating systems. On most modern machines, similar basic recording systems of tracks and sectors are used on all systems from BBC micros through CP/M micros to IBM PC's and the compatibles running MSdos. However, it must be said that Apple 5.25in drives are different from anybody else's system.

Furthermore, the sectors are not used consecutively in a physical sense, rather they are logically formatted. This has nothing to do with the way in which data is actually written on the disc, but rather the way sectors are grouped together by an operating system to create files. Thus different systems have different logical formats.

This is clearly shown by the various disc operating systems in common use in the Apple 5.25in drives. You know that COPYA or the Prodos filer will copy the whole disc, be it Dos, Prodos, Pascal or CP/M, but each can only copy individual files under its own system.

Logical formats also control the method used to build a directory, the group of sectors which hold an index of the files on the disc and which point to the locations of each.

CP/M is designed to run on any Z80 or 8080 based computer and therefore has to be immune to the chaos caused by the vast variety of physical disc formats used on different micros. To make this possible the BIOS – the machine specific part of CP/M – does all the worrying about physical format.

CP/M – and any program written to run under CP/M – treats all discs as having an idealised sector size of 128 bytes. Thus as far as CP/M is concerned the Apple 5.25in disc has 32 sectors of 128 bytes per track, and the BIOS has to translate this into the physical layout of 16 sectors of 256 bytes per track whenever CP/M wants to read or write a 128 byte sector from or to disc.

The process of buffering carried out by the BIOS to do this is known as blocking and deblocking. We won't go into any more detail here because this is transparent to the user.

To avoid confusion, if we haven't already, we'll refer to CP/M's 128 byte logical sectors as records because this is what CP/M calls them. We'll treat our disc as having 32 records per track.

On a system disc, tracks 0, 1 and 2 are reserved to hold the code which loads CP/M and the code which makes up CP/M itself and are therefore not used to hold data. The directory is held in the first 12 records on track 3.

This is a bit like the contents page in a magazine – it tells us where to look in the disc to find a particular file of data or a program. To reduce the amount of information the directory needs to hold, and also its size, CP/M imposes an extra level of organisation on itself by grouping records together into blocks. Under the Microsoft v2.2 system eight records are grouped into one 1k block.

These blocks are an abstract concept and invisible to programmers in the normal course of events: They just exist to make things easier for CP/M. However, we have to be aware of their existence to understand how the directory is constructed.

Each record in the directory contains four 32 byte File Control Blocks, each of which takes the form shown in Figure I. Each FCB contains information on the location on the disc of one part, or extent, of a file.

The first byte contains the user number of the file and is therefore usually 0 on an

Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Contents	user	F	I	L	E	N	A	M	E	T	Y	P	ex	0	0	rc
Byte	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Contents	d0	d1	d2	d3	d4	d5	d6	d7	d8	d9	d10	d11	d12	d13	d14	d15

Figure I: Contents of a directory File Control Block

```
A: TURBO    MSG : TURBOMSG OVR : TURBO    OVR : TLIST    COM
A: TURBO    COM : TINST    DTA : TINST    MSG : TINST    COM
A: PIP      COM : STAT     COM : COPY     COM : TEPSON   COM
A: TIMAGE   COM
```

Figure III: Directory of a disc shown by DIR

Apple CP/M system running 5.25in drives. A value of $E5 means that the file has been deleted. When CP/M deletes a file it doesn't actually wipe out the contents of the file on disc, it simply sets the user byte in all the file's extents to $E5.

This allows us to write clever little programs called UNERASE.COM to recover files we've accidentally blown away. This is clearly seen by comparing Figure II, which is the directory of a disc as shown by the DIR command, with Figures IV-VIII, which show the first five records containing the directory of the same disc. Note that the files POLYNOM.PAS and JOHN1.PAS have been deleted in Figure VIII.

Bytes 1 to 8 of the FCB (Figure I) contain the filename in Ascii upper case (note that the high bit is reset as opposed to Apple Dos where the high bit is set) and bytes 9 to 11 hold the file type, again in Ascii upper case. Note that the . we normally put into FILENAME.TYP is not put into the directory.

Bytes 12 and 15 are the extent number and record count values respectively, we'll return to these later. Bytes 13 and 14 are always 0; they are reserved for internal use by the BDOS during file during file operations.

Bytes 16 to 31 are the index, which tells us where to find the data for the file. Each byte is the number of a block which has been allocated to the file.

As we said previously, Microsoft CP/M on the Apple uses a 1k block size. Each FCB has 16 bytes reserved for holding allocation information and so can map up to 16k of a file into each of its extents.

If the file is 16k or less then obviously it only requires one extent, and its extent number (byte 12 of its FCB) is set to 0. If the file is larger than 16k it will need more than one extent to hold its directory information and subsequent extents will be numbered 1, 2 and so on.

Byte 15 of the FCB, the record count, is the exact number of records which contain valid data held within the blocks pointed to by that extent. As an extent can hold up to 16k and a record is 128 bytes in size the maximum number of records held by an extent is obviously 16x8 or 128. Thus the extents of large files which are completely filled will have record counts of $80. If the last extent is not completely filled it will have a count less than this.

This may be clarified by comparing Figure III, which shows the information derived by STAT from our disc, and Figure IV, which shows that the file TURBO.MSG has its entry in one extent because it is only 12 records long. Note that STAT says it is 2k long but is really only 1.5k. Also compare STAT's information in Figure III with the information in Figure V which shows the second record of the directory. Note that TURBO.COM occupies two extents in its directory entry (it is 30k long) and has $80 records in the first and $6E in the second which add together to give $EE or 238 records in total.

You may well come across terms in CP/M literature such as extent folding and information, which suggests that the structure of extents is more complex than we have described here. Unfortunately that is so for some systems. The problem is that most computers use different logical formats on their discs and many have block sizes larger than 1k.

```
stat a:*.*

 Recs  Bytes  Ext Acc
   28     4k    1 R/W A:COPY.COM
   58     8k    1 R/W A:PIP.COM
   48     6k    1 R/W A:STAT.COM
   67     9k    1 R/W A:TEPSON.COM
   66     9k    1 R/W A:TIMAGE.COM
  195    25k    2 R/W A:TINST.COM
   35     5k    1 R/W A:TINST.DTA
   30     4k    1 R/W A:TINST.MSG
  119    15k    1 R/W A:TLIST.COM
  238    30k    2 R/W A:TURBO.COM
   12     2k    1 R/W A:TURBO.MSG
    8     1k    1 R/W A:TURBO.OVR
   11     2k    1 R/W A:TURBOMSG.OVR
Bytes Remaining On A: 6k
```

Figure II: Directory of a disc shown by STAT

```
Track $03/Sector $00/
$00  00 54 55 52 42 4F 20 20  _TURBO
$08  20 4D 53 47 00 00 00 0C   MSG____
$10  02 03 00 00 00 00 00 00  ________
$18  00 00 00 00 00 00 00 00  ________
$20  00 54 55 52 42 4F 4D 53  _TURBOMS
$28  47 4F 56 52 00 00 00 0B  GOVR____
$30  04 05 00 00 00 00 00 00  ________
$38  00 00 00 00 00 00 00 00  ________
$40  00 54 55 52 42 4F 20 20  _TURBO
$48  20 4F 56 52 00 00 00 08   OVR____
$50  06 00 00 00 00 00 00 00  ________
$58  00 00 00 00 00 00 00 00  ________
$60  00 54 4C 49 53 54 20 20  _TLIST
$68  20 43 4F 4D 00 00 00 77   COM___w
$70  07 08 09 0A 0B 0C 0D 0E  ________
$78  0F 10 11 12 13 14 15 00  ________
```

Figure IV: First record of a disc directory

```
Track $03/Sector $03/
$00  00 54 49 4E 53 54 20 20  _TINST
$08  20 4D 53 47 00 00 00 1E   MSG____
$10  39 3A 3B 3C 00 00 00 00  9:;<____
$18  00 00 00 00 00 00 00 00  ________
$20  00 54 49 4E 53 54 20 20  _TINST
$28  20 43 4F 4D 00 00 00 80   COM____
$30  3D 3E 3F 40 41 42 43 44  =>?@ABCD
$38  45 46 47 48 49 4A 4B 4C  EFGHIJKL
$40  00 54 49 4E 53 54 20 20  _TINST
$48  20 43 4F 4D 01 00 00 43   COM___C
$50  4D 4E 4F 50 51 52 53 54  MNOPQRST
$58  55 00 00 00 00 00 00 00  U_______
$60  00 50 49 50 20 20 20 20  _PIP
$68  20 43 4F 4D 00 00 00 3A   COM___:
$70  56 57 58 59 5A 5B 5C 5D  VWXYZ[\]
$78  00 00 00 00 00 00 00 00  ________
```

Figure VI: Third record of a disc directory

This means that each extent can then in theory index more than 128 records and we have to fold logical extents into each physical extent. Yes, our eyes start to glaze too. Fortunately, on the Microsoft Apple system we don't have to worry about this sort of thing.

We said earlier that the directory is 12 records long, or 1.5k in size. This means that it can only hold 4x12 or 48 entries. Note that this is the maximum number of byte entries – that is, extents, not files. If the directory gets full we cannot put anything else on to the disc until space has been made by deleting something.

This is a separate problem from the more common one of simply running out of data space on disc. The directory size of 1.5k is an arbitary one determined by Microsoft when it considered how best to use Apple's disc format. Other CP/M computers use different sizes of directory, for example 64 entries in 2k as on the Amstrad machines or commonly 128 entries in 4k.

So that programs can find out information like this about the system on which they are running the BIOS maintains a table of data called the Disc Parameter Block or DPB. This contains information on our logical disc format and is accessed by executing a BDOS function call 31 which returns the base address of the DPB in the register pair HL. We'll be looking at the disc function calls next month.

Figure IX lists the contents of the DPB under Microsoft v2.2 by the shortened mnemonics by which they are known. The ▷

```
Track $03/Sector $00/
$80  00 54 55 52 42 4F 20 20  _TURBO
$88  20 43 4F 4D 00 00 00 80   COM____
$90  16 17 18 19 1A 1B 1C 1D  ________
$98  1E 1F 20 21 22 23 24 25  __ !"#$%
$A0  00 54 55 52 42 4F 20 20  _TURBO
$A8  20 43 4F 4D 01 00 00 6E   COM___n
$B0  26 27 28 29 2A 2B 2C 2D  &'()*+,-
$B8  2E 2F 30 31 32 33 00 00  ./0123__
$C0  1F 63 70 2F 6D 20 20 20  _cp/m
$C8  20 73 79 73 00 00 00 60   sys___`
$D0  80 81 82 83 84 85 86 87  ________
$D8  88 89 8A 8B 00 00 00 00  ________
$E0  00 54 49 4E 53 54 20 20  _TINST
$E8  20 44 54 41 00 00 00 23   DTA___#
$F0  34 35 36 37 38 00 00 00  45678___
$F8  00 00 00 00 00 00 00 00  ________
```

Figure V: Second record of a disc directory

```
Track $03/Sector $03/
$80  00 53 54 41 54 20 20 20  _STAT
$88  20 43 4F 4D 00 00 00 30   COM___0
$90  5E 5F 60 61 62 63 00 00  ^_`abc__
$98  00 00 00 00 00 00 00 00  ________
$A0  00 43 4F 50 59 20 20 20  _COPY
$A8  20 43 4F 4D 00 00 00 1C   COM____
$B0  69 6A 6B 6C 00 00 00 00  ijkl____
$B8  00 00 00 00 00 00 00 00  ________
$C0  00 54 45 50 53 4F 4E 20  _TEPSON
$C8  20 43 4F 4D 00 00 00 43   COM___C
$D0  64 65 66 67 68 6D 6E 6F  defghmno
$D8  70 00 00 00 00 00 00 00  p_______
$E0  00 54 49 4D 41 47 45 20  _TIMAGE
$E8  20 43 4F 4D 00 00 00 42   COM___B
$F0  71 72 73 74 75 76 77 78  qrstuvwx
$F8  79 00 00 00 00 00 00 00  y_______
```

Figure VII: Fourth record of a disc directory

```
--------------                 ---------------
Track $03/Sector $06
----------------------------------------------
$00   E5 50 4F 4C 59 4E 4F 4D   ePOLYNOM
$08   20 50 41 53 00 00 00 20    PAS___
$10   7A 7B 7C 7D 00 00 00 00   z{|}____
$18   00 00 00 00 00 00 00 00   ________
$20   E5 4A 4F 48 4E 31 20 20   eJOHN1
$28   20 50 41 53 00 00 00 04    PAS____
$30   7B 00 00 00 00 00 00 00   {_______
$38   00 00 00 00 00 00 00 00   ________
$40   E5 E5 E5 E5 E5 E5 E5 E5   eeeeeeee
$48   E5 E5 E5 E5 E5 E5 E5 E5   eeeeeeee
$50   E5 E5 E5 E5 E5 E5 E5 E5   eeeeeeee
$58   E5 E5 E5 E5 E5 E5 E5 E5   eeeeeeee
$60   E5 E5 E5 E5 E5 E5 E5 E5   eeeeeeee
$68   E5 E5 E5 E5 E5 E5 E5 E5   eeeeeeee
$70   E5 E5 E5 E5 E5 E5 E5 E5   eeeeeeee
$78   E5 E5 E5 E5 E5 E5 E5 E5   eeeeeeee
----------------------------------------------
```

Figure VIII: Fifth record of a disc directory

Byte		Contents	Value	
1,0	SPT	Records per Track	32	$20
2	BSH	Block Shift Factor	3	$03
3	BLM	Block Mask value	7	$07
4	EXM	Extent Mask value	0	$00
6,5	DSM	Total Storage Capacity	139	$008B
8,7	DRM	Max. Directory Entries	47	$002F
9	AL0	Directory Blocks	192	$C0
10	AL1	Directory Blocks	0	$00
12,11	CKS	Checked Records	12	$0C
14,13	OFF	Reserved Tracks	3	$03

Figure IX: Contents of a Disc Parameter Block (DPB)

◁ values of each are held in one byte or in two bytes in which case the second byte is generally more significant. This is what they all mean.

SPT is Sectors per Track. This refers to CP/M's logical sectors of 128 bytes, i.e. records, and not our real physical sectors of 256 bytes. As we saw earlier CP/M considers us to have 32 ($20) records per track.

BSH is Block SHift factor and this and BLM (BLock Mask) are concerned with the size of our data blocks – 1k each in the Apple. This requires a Block Shift factor of three and a Block Mask value of seven, but we are not going to explain how these are derived here.

EXM is the EXtent Mask and is used to control the extent folding mentioned earlier. On the simple Apple system this is zero.

DSM is one less than the total storage capacity of the disc in units of blocks. So for us to have 32 records (or 4k per track) times 35 tracks allows 140 blocks of 1k so the DSM is 139 ($8B).

DRM is one less than the maximum number of directory entries. We saw earlier that this is 48 and so DRM is 47 ($2F).

ALO and **ALI** make up a 16 bit word (with ALO as the more significant byte) which tells us how many blocks are allocated to the directory. It does this by setting a bit of the word, starting at the most significant end of ALO for each block allocated. We use two blocks, 0 and 1, so the word value is binary 11000000 00000000 which in hexadecimal is $C000.

CKS is CheckEd Sectors which tells CP/M the number of directory records which must be read and checked when a disc is accessed to discover if a different disc had been inserted into the drive without being logged in – the cause of the infamous BDOS ERR ON A. Obviously with simple removable floppy discs such as ours we would need to check the entire directory on a disc to be able to be sure that it has not changed and so we have a CKS value of 12 ($0C) records.

The last entry in the standard DPB is **OFF** which is the OFFset. This tells CP/M the number of tracks which we have reserved on our discs to hold the code for CP/M itself – three in our case. □

Upgrade now . . . to the Apple dream machine

MORE and more Apple User readers have a Macintosh – and many more Apple II users are now thinking of upgrading to the more sophisticated Apple machine. To help stimulate this interest we invite all our readers to take part in this month's competition.

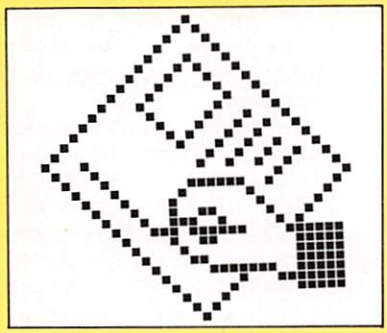
The prize: a Mac Plus worth £1,995, kindly donated by Apple UK. PLUS a copy of Icon Technology's MacAuthor – the soon to be released version 1.3, worth £199 – and, from Rainbird, a copy of The Pawn and its follow-up, Guild of Thieves.

What you have to do

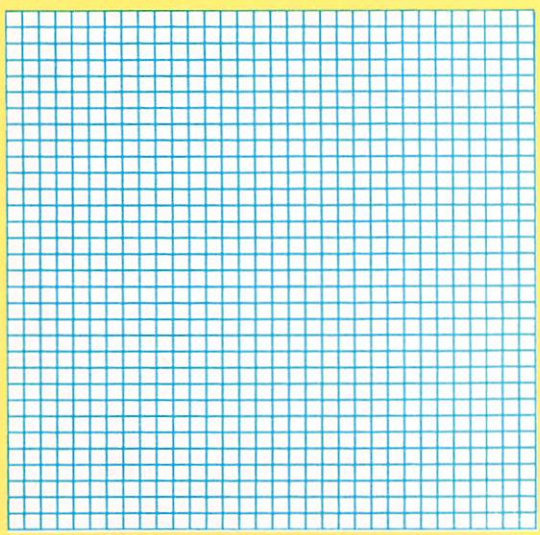
Your task is to design suitable icons for three new software packages: a word processor, database and spreadsheet. The Macintosh's icons are contructed using a 32x32 grid, so to help you we have provided a sample blank below.

You can copy this, use graph paper, or draw freehand – providing you bear in mind the limitations of the grid. You may want to put a name to each package, but it is not essential.

Fill in the entry form below and attach it, or a photocopy, to your designs. Send your entry to the address below to arrive no later than July 31, 1987.



The Pagemaker icon viewed in FatBits

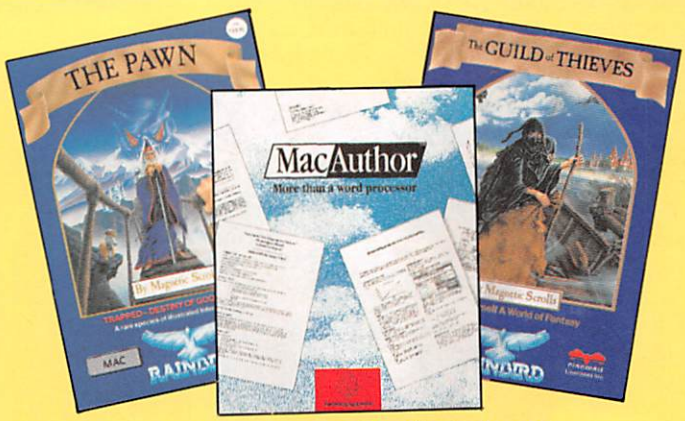


Sample of 32 × 32 grid

The Macintosh Plus comes complete with a 78-key keyboard, precision-engineered mouse, 800k internal drive, comprehensive user manual and system discs.

It features:

- 512 x 342 resolution, nine inch display
- One megabyte of ram and 128k rom
- 256 bytes of battery-backed ram
- MC68000 32-bit processor, 7.8336 MHz
- 4-voice sound with 8-bit digital analog conversion using 22 KHz sample rate



The three software titles also to be won

Entry form

Name_____

Address_____

_____

_____

Postcode_____

Send to: Macintosh Competition, Apple User, *Europa House, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.*

Program: Bureaucracy
Price: £34.99
Supplier: Infocom c/o Activision,23 Pond Street, London NW3 2PN.
Tel: 01-431 1101
Requirements: Apple Macintosh or Apple II with 64k.

It's fun being well wrapped up in red tape

ONCE upon a nightmare, a man went to his bank to tell them that he had just moved house and would they please note his new address. He should have known better.

The bank would not accept the notification because it had not been completed on their own special change-of-address form and the computer would not like that. Not to worry, said the bank, we'll send you the correct form through the post – just fill it in, return it to us and everything will be just hunky-dory.

Guess what? The bank sent the blank to his old address. Unsuspecting and naively innocent, the man tried to use his credit card and found it had been invalidated. Guess what again? The bank had cancelled his credit card and reissued a new one, sending that to his old address.

The man tried talking to the bank, filling in oodles of new forms and endeavoured to get a new credit card issued, but all to no avail. The poor wretch was yet another victim of that curse of the civilised world – Bureaucracy.

If you've ever been caught up in the sticky coils of red tape, whether it be by bank, postal service, airline, government, telephone company or any other known generator of the scarlet stuff, then you're going to love this new, painfully funny, Infocom text adventure.

And I mean painfully funny. Many of the situations in the adventure could be all too familiar to your own experiences and may well cause the blood pressure to rise and the ulcer to throb. However, Bureaucracy is also very, very funny and just as likely to have your face aching with smiling at its wit and humour.

Which is what you might expect since that master of the comically absurd, Douglas Adams, played a major part in writing it.

Bureaucracy begins by warning you that you do not have a licence to operate the software. In order to avoid a wait of about six months while a postal application is processed, you may continue to play provided you complete an online, electronic application form.

And what a masterpiece of bureaucratic computer form design – the cursor leaps wildly from field to field with no logical pattern.

Among the questions you'll be asked are the name of your previous and current boy/girl friend and your least favourite colour. The immediate response to your completion of the form might lead you to think that there's a bug in the program since you are initially addressed as being of the opposite sex.

It's no bug, just an early example of bureaucratic bungling to get your blood pressure up. Untypically, this particular blunder is soon rectified. And much of the information you feed into the form is used to splendid effect later in the game.

Your role in the adventure is that of a new employee of Happitec Corporation, who are sending you on a two-week, all-expenses-paid trip to Paris. There you'll attend a training seminar but will still have lots of time and cash for fun and leisure. A money order has been sent to your new home and you can pick up the airline tickets at your travel agency.

What a benevolent employer! Well, maybe not – there's a sting in the tail of your instructions. "Be sure to be on time for your flight. We've found that new employees who miss the training scheme usually aren't very happy at Happitec." Point taken.

Two numbers are shown at the top right edge of the screen indicating the state of your blood pressure. Every time you enter an action that is of no use, or whenever you encounter frustrations in your dealings with people and events, it soars: It descends when life is calmer. Should the reading go too high, you expire and must restart (or restore a saved game). The Score command will access your marks (out of 21), blood pressure status and current rating.

The game begins with you in the front room of your new apartment. The place is pretty bare because the removal firm have slipped up over the dates. A delivery man soon turns up but he doesn't have your money order. What he does have is a bag of llama treats which, according to his records, you ordered.

If you want this bizarre package, it's payment on delivery – he'll take Excess or Beezer credit cards which you just happen to have in your wallet. (By the way, what is the largest known number in the universe? Yep, your credit card number).

Your back room has a few items in it, among which are a computer and a telephone-cum-answering machine. The computer is a Boysenberry (with a partially-digested purple berry as its logo) on which you can run an adventure or an eclipse

prediction cartridge. The adventure (DORK I) soon crashes out with an internal error message (Boysenberry products are clearly unreliable. There are a couple of other computers referenced elsewhere in Bureaucracy – a Daktari SM and a Cormorant Honcho.

I strongly recommend that you listen to the answering machine and dial several telephone numbers – this had me laughing so much I did it all over again. One of the delights of Infocom adventures – and Bureaucracy in particular – is the sheer amount of thought, wit and detail that has been lavished on the program. You'll get an entertaining response from far more of your inputs with Infocom than any other adventures you care to name.

Rampant bureaucracy awaits you outside your apartment. The Fillmore Fiduciary Trust Bank will have you tearing your hair out as you move from one to another of their 10 teller windows, trying to change your address, make a deposit or withdrawal or cash a cheque.

In the Travel Agency, the agent sits waiting to misunderstand where customers want to go and how to get there just so she can earn a fat commission by sending them to Djakarta. She's the height of tact and diplomacy – show her your passport and she'll say "Visa's fine but, hey, what kind of happened to your face, buster?" Well, you were cautioned earlier when examining your passport photo that "you wonder if you really do look like a dead llama".

The fast food restaurant is inaptly named for the food is not fast (ordering is a complicated process) nor is it food (small, wrongly-cooked, dog-eared burgers). The waiter who eventually takes your order is the type who kicks whole beaches in people's faces. He has small simian eyes, hands like huge hairy hams and the intellectual effort of reading his own writing puts a great strain on his mental tackle.

The mansion house is particularly interesting. In it you'll find a macaw with its left wing missing, who screams out "Nuke the whales!" and other objectionable slogans – a right wing parrot! In the nearby trophy room, a dowdy matron sits with her ear trumpet glued top the radio. In her lap nestles an elephant gun.

The adventure is crammed with dry and witty humour. Examine the modular jack that comes with the computer, for example, and you'll be told "It is modular and it is a jack. Hence the name modular jack". In Zalagan currency, a particular stamp is worth many Zalagasan Wossnames – "they were too idle to think of a name".

As to be expected with Infocom products, the packaging is immaculate. In it you'll find the letter from Happitec, some friendly advice from your bank, an advertisement for Popular Paranoia magazine (Did you know that cash dispensing machines are used to keep track of your movements?), and a tiny, thin pencil for completing your credit card application form. This form is in triplicate (make sure you check out the copies) and as well as being a bureaucrat's dream, is genuinely funny to read.

There's no doubt that this new Infocom adventure is a worthy successor to Hitch Hiker. It combines all the professionalism and quality that Infocom bring to their games with the zany and entertaining style of Douglas Adams' writing.

What is certain is that Infocom will score a direct hit with their public – Bureaucracy is as fresh, fertile and funny an adventure as ever crackled across your screens. And you can take that from me in triplicate.

Bob Chappell

Program: Jewels of Darkness
Price: £19.95
Supplier: Rainbird Software, First Floor, 74 New Oxford Street, London WC1A 1PS.
Tel: 01-631 3589
Requirements: Macintosh or Apple II with 64k (colour card optional)

Triple winner in the darkness

JEWELS of Darkness is an adventure trilogy of such quality and at such a bargain price that it should bring a smile to the faces of all Apple adventurers. Here we have three excellent adventures, handsomely packaged and including an entertaining 64 page novella, The Darkness Rises, and a 12 page guide to the games. All this for virtually the price of one game.

The adventures are not new, but are among the first three released by Level 9 a few years ago for the less expensive home micros: Colossal Adventure, Adventure Quest and Dungeon Adventure. And this trilogy is made of the stuff that adventurers crave, make no mistake. Colossal Adventure is an adaptation of the original Crowther & Woods classic, Colossal Cave. And Level 9's version is, in my book, the best ever. While remaining faithful to the original (it contains all the treasures, creatures, rooms and puzzles of the Crowther & Woods masterpiece), it introduces an additional end-game, continuing the adventure where the original left off.

In this fantasy, you are on the trail of fabulous treasures hidden in a massive, awe-inspiring cave. Points are scored for finding the cave, gathering the treasures (with many puzzles to be solved along the way) and returning them to the famous small brick building which is situated near the beginning of your journey. Points are lost if you are killed (you might get lucky and be brought back to life in a puff of purple smoke for another attempt).

Colossal Adventure features many marvellous descriptions and incidents and is one of the best adventures ever written – no adventurer worthy of his or her brass lamp and elvin sword should be without a copy.

Adventure Quest has you playing the role of an apprentice magician pitted against the Demon Lord, Agaliarept, whose ▷

◁ vast armies are sweeping all before them.

You are summoned before the Wizard's High Council and tasked with finding the four Stones of the Elements – they alone are the key to entering the Black Tower, Agaliarept's headquarters. Once there, the Amulet of Life could be your one chance of defeating the Demon Lord.

A treasure hunt is the central theme of Dungeon Adventure in which, without weapons or magic, you enter the ill-famed, monster-filled dungeons of the Demon Lord.

As good as these adventures were when they first saw the light of day, Rainbird have not been satisfied with simply reissuing old material. Each of the programs has been subjected to a thorough rewrite, changing the overall style yet, thankfully, leaving the text faithful to the original classics.

A superb new parser has been incorporated into each adventure, allowing far more complex, multi-command input. The vocabulary has been extended to over 1,000 words and the trilogy sports more than 600 colourful illustrations for those who prefer a helping of graphics with their adventuring. The simple, stylised pictures are nowhere near the standard of The Pawn, but they are bright and plentiful and are drawn very quickly.

The response-time to your typed-in commands is almost instantaneous, which makes the adventures a real pleasure to play. And the text messages have been greatly expanded to give fuller, more richly detailed descriptions.

Jewels of Darkness is a superb trilogy sumptuously packaged and a joy to play. It would be worth the asking price for Colossal Adventure alone, but with two other excellent adventures included for the money, Jewels of Darkness is unquestionably the Apple bargain of the year.

Bob Chappell

Program: AutoDuel
Price: £24.95
Supplier: Origin Systems c/o Microprose, 2 Market Place, Tetbury, Gloucestershire.
Tel: 0666 54326
Requirements: Any 64K Apple II and a joystick.

Take to the road the Elite way

AUTODUEL is the product of Lord British of Ultima fame, and Chuckles who wrote the legendary Caverns of Calisto. Based on Steve Jackson's board game Car Wars, AutoDuel is a futuristic game of survival and power which centres around the player's car.

The game begins with a full screen plan of the city. You can move your little person around the screen and try and enter various buildings. You start with only a small amount of cash, and the first place to visit is the Arena. On checking the roster you discover that it is Amateur night, and it's usually a good idea to enter – it costs nothing anyway.

You are loaned a car in which to battle, along with a somewhat limited supply of ammunition. The idea is to drive around the full-screen, scrolling course and kill a certain number of enemy cars. Once you accomplish this, you are awarded a generous cash prize.

After you have accumulated a fair amount of cash, you can go about building a decent car. There are all sorts of options to choose from. First there is the type of chassis – anything from a light frame to a rugged heavy duty one. Then you can add various engines, armour plates, tyres, weapons and so on to your car.

At all times you need to keep everything in proportion, otherwise you will end up restricting some vital feature on your car, like cargo space, or making it too heavy.

Having built a decent car, you can do various things with it. Parking it in the garage will cost you a certain amount per day, but it makes the time pass to the right day for the competition you want to enter at the Arena. There are various divisions, with each more difficult and more rewarding than the last.

If it gets damaged or needs fuel (battery charging) or extra ammunition, you can take the car to get it fixed – at a hefty cost to your bank balance. You can even get yourself repaired if you come to any grief too. There are bars, shops and many other types of building to be found in the various cities, but the number and type in each city depend on its size.

The other way to make your fortune is to carry precious cargo from one city to another. A list of jobs is available, which tells you what objects have to be delivered, where, by what date and, most importantly, how high your reward is.

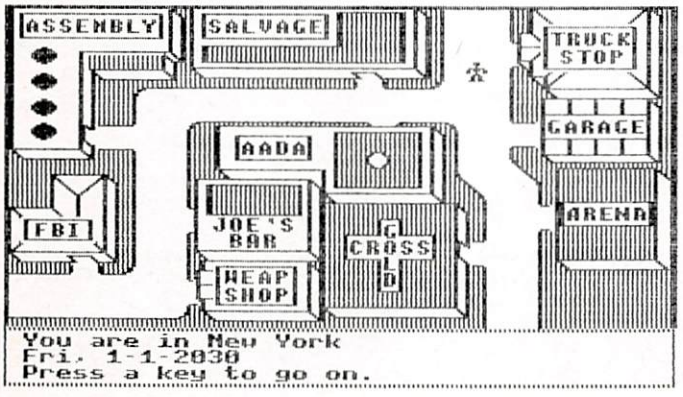
Various objects take different amounts of cargo space, which needs to be taken into account. Naturally the less time you take and the further you have to deliver an object, the higher the reward. If you find yourself unable to complete the mission, you can always sell the item to another punter for a fair price.

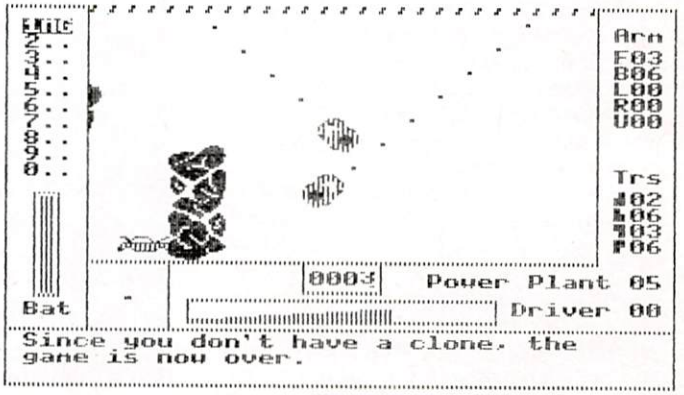
The roads that link the various cities are pretty long, and you will usually meet pirate drivers along the way – and engage them in battle. Once they're disabled, you can get out of your car and search the wreckage for cargo, spares, money and so on. This is a good – but potentially hazardous – way to get rich.

The games come with a map of North America showing all the cities and the highways which link them, as well as the facilities within each city.

AutoDuel is a challenging and absorbing game, rather like a car-based Elite, with similar elements – trading, missions, accumlation of weapons and money.

Leon Seltsikas





The World of Communications for Your Apple Macintosh

VICOM (AM Technology) £150

With your Macintosh connected through a Modem to the telephone system, you can dial up any of the large database systems; as many as available on any other PC. These range from Prestel, Telecom Gold, Mercury, One-to-one, etc where messages can be left, telex replied to and large amounts of information referred to. Other main-frame computers can also be accessed.

MAC LINK PLUS (DataViz) £195

Using the Cable provided with this package, you can connect your Macintosh directly to an IBM or IBM compatible, through the serial port on each machine. Put the 5.25" disk in the IBM and the 3.5" disk in the Mac. Now you can transfer any file from the IBM to the Mac and vice-versa. Furthermore, the Mac Link software is able to convert files between the formats used by popular programs on both machines.

TOPS (Centram) £149

Macintosh Version (per user)

With an Apple Talk network connecting a number of Macintosh computers, TOPS runs as a file server, allowing files on hard disks attached to one or more of these machines to be accessed by any of the others. Full Read/Write control is provided.

TOPS (Centram) £425

IBM Version (per user)

Now plug the TOPS AppleTalk card into your IBM, or compatible; connect up to the Apple Talk Network, and you can load and save files held on your IBM hard disk with one of the Macs equipped as above. For example, open Lotus 123 files (stored on your IBM) on your Mac using Excel. Edit the spreadsheet, then save it back to the IBM. Magic! Print from Microsoft Word V3.0 on your IBM using the same AppleTalk network connection to the LaserWriter.

Call in or telephone us for more information on any of these products, or other hardware or software for the Apple Macintosh.

INTERMAIL (Internet) £249

(1-5 User Version)

Apart from sharing the use of an Apple LaserWriter over AppleTalk, Intermail uses this network to send messages and files between users on the network. Simple commands for receiving or sending information, even from within other programs, makes this a very useful E-Mail package.

INTERBRIDGE (Hayes) £799

Attach this box to your AppleTalk system, and then to a Hayes SmartModem, running at 2400 Baud. This allows you to connect up AppleTalk systems in different locations using telephone lines, without losing any of the flexibility of the products listed above. For example, send Intermail messages, with Excel spreadsheet files attached, to different executives in different parts of the country, directly to the Macintosh on their desk. If they are out, the message/file waits until they return, before being delivered. Print reports on a LaserWriter in a different location.

N 1000A (Tri-Data) £2565

(SNA or Bi-Sync Version)

Connect your AppleTalk network to an SNA link on an IBM Mainframe. This box simulates an IBM 3274 cluster controller, allowing up to 8 Macintosh computers to carry out simultaneous sessions as 3278 terminal emulators to the main frame. Full cut & paste of screen information back into the Macintosh environment.

FASTPATH GATEWAY/ ALISATALK (Kinetics) £7500

(VAX 11/785 or 8300 Version)

Connect your AppleTalk network to a VAX, using Ethernet. Allows the Vax to act as a Network file-server for the Macs on the AppleTalk network. Also provides network access to the VAX and printer spooling.

* ICL and IBM System 34/36 links now also available.

(Prices exclude VAT and are subject to confirmation).

STIRLING MICROSYSTEMS

The Specialist Centre for Apple Macintosh Systems

231 Baker Street, London NW1 6XE. Tel: 01-935 5262

Open: Monday to Friday, 9.30 to 6.00

Authorised Apple Dealer

Enhanced machine code tracer

Trevor Hobson offers a revised version of a useful machine code utility

THE March 1986 edition of *Apple User* contained a program called Step by Step by Michael Agerbak, for tracing machine code programs.

I have been using it over the past few months and have found it an extremely valuable tool for "single stepping" through machine code programs and observing the effects on the 6502 registers and on selected memory locations.

However, I did question some of the things which it was doing, and I have just carried out a significant revision of the original program. Although I have not modified the structure, the result is a program which contains several major enhancements – and fewer bugs. The revised program is shown in Listing I.

In Listing II I have included a test program containing some of the program constructs which I came across while using the original program. If any of you have already keyed in the original, and wonder whether it will be worth the effort to key in this revision, then I suggest you attempt to trace the test program in Listing II using the original version.

I'm afraid you will not get very far, and even changing the page-0 locations used by the test program will not help very much. The revised version traces this test program successfully.

The program in Listing II provides several enhancements, and the major ones are listed below.

Use of page zero: In the original program any attempt to share a page-0 location with the program being traced would usually lead to some interesting but certainly unwelcome side effects.

The revised program can share any page-0 location with the program being traced. It does this by storing a copy of page-0 elsewhere in memory. This is periodically exchanged with the existing contents of page-0 so that the correct version is seen by both the tracing program and the program being traced.

It is now possible to have two copies of Step by Step Version 2.0 assembled at different locations in memory, and to use one copy to trace the other while it is tracing yet another program. What is more, all three programs could be sharing the same page-0 locations.

Trapping calls to rom: The original program correctly traps all JSRs to rom routines and offers you the option not to execute them. There are, however, other ways that a rom routine could be accessed by the program being traced, which are not trapped by the original program. These are:

- By executing a JMP to a rom address (some programmers use this at the end of a subroutine in preference to a JSR followed by an RTS).
- By pushing a rom address into the stack and then executing an RTS instruction.

Also, from what I have seen of the way some people write programs, it would not surprise me to find that somebody out there has contrived to execute a Branch to a rom routine. Not very practical, but strictly speaking not impossible.

The revised program will trap all attempts to execute an instruction in rom, however it is caused.

Change the contents of memory locations: The revised program provides an additional command which enables the contents of specified ram memory locations to be changed. Changes to page-0 or page-1 (stack) locations are handled as you would expect, with the change being made to the copy of page-0 or page-1 "seen" by the program being traced.

Changes to memory locations can be made in either the Editor or the Tracer, by pressing C, then entering the address of the memory location to be changed. Press Return, enter the value to be stored in that location and press Return again.

For ease of reference Table I provides a list of all the commands available with Step by Step version 2.0.

Display of page-0 and stack locations: When using the D command (to display the contents of a specified memory location), if either a page-0 or page-1 (stack) location is specified then the revised program will automatically display the values that relate to the program being traced. That is, the program will automatically display the contents of the correct memory location in either the copy of page-0 or in the simulated stack as appropriate. The existence of the copy of both page-0 and page-1 is thus entirely invisible to you.

Improved screen display: The original program would execute some instructions without your being able to see what the instruction was until after it was executed. In addition, the status of the top line of the disassembled listing was not clear:

Sometimes, but not always, it was the instruction which had just been executed. Finally, executing some of the program's commands would cause the disassembled listing to scroll off the top of the screen.

```
0917–  20 2B 09     JSR   $092B   line just executed
092B–  A5 0C        LDA   $0C     line about to be executed
092D–  20 ED FD     JSR   $FDED
0930–  20 8E FD     JSR   $FD8E
0933–  60           RTS
0934–  A5 0D        LDA   $0D
0936–  4C ED FD     JMP   $FDED

       NV-BDIZC
P      10110000     B0    contents of 6502 registers
S      11111101     FD    contents of 6502 registers
A      11000101     C5    contents of 6502 registers
X      11010001     D1    contents of 6502 registers
Y      10110001     B1    contents of 6502 registers

       (01FE):19           display of a memory location
```

Figure 1: Sample Screen Display

Figure 1 shows a sample screen layout produced by the revised program. The top line of the screen is always the last instruction which was executed in the program being traced. Then there is a blank line followed by six lines of disassembled program, the top line of which is always the instruction to be executed.

Beneath the lines of disassembled progam is the display of the contents of the 6502 registers, and then (in brackets) the address of the memory location selected for display by the D command, followed by the current value stored in that location.

The whole of this display remains firmly in place even when the program's commands are executed.

Simulation of RTI: In the original article Michael Agerbak said that the program did not accommodate the instructions RTI or SEI. While he was quite correct about RTI, the original program did in fact correctly handle SEI.

The revised program includes code to simulate operation of the RTI instruction. All documented 6502 op-codes will therefore now be traced by the revised program.

Improved control over auto-repeat and break point: When using the auto-repeat function (command F) and the break point function (command B), the original program would return to the Editor once the break point address was reached by the Tracer but would not cancel either of these two functions.

As a result, if you wish to continue tracing in single step mode from where the auto-repeat mode had left off it was necessary to reset the break point address manually, return to the Tracer and quickly press the spacebar to cancel the auto-repeat function before the Tracer had chance to progress too far.

The revised program cancels both the auto-repeat and the break point functions when control is returned to the Editor, giving an improved and simpler control over these functions.

Trapping invalid input: In the revised program the only input allowed to the question "EXECUTE?" is Y or N. Nothing else will be accepted.

Simplify relocation: The program in Listing I was assembled at $0B00. To assemble at another location it is necessary only to change the address of the ORG instruction and then re-assemble. In the original it was necessary also to change the value of the label STACK. I for one kept forgetting to do this, sometimes with very strange consequences.

Simulate 6502 JMP() bug: The 6502 microprocessor contains a bug (Ref: page 151 of 6502 Assembly Language Subroutines by L. A. Leventhal and W Saville, Pub: Osbourne/McGraw-Hill). In short, if the instruction JMP(XXFF) is encountered the processor does not take the destination address from page XX byte FF and page XX+1 byte 00 as you might expect, but instead takes the destination address from page XX byte FF and page XX byte 00 (that is, the page number is not incremented when getting the low byte of the destination address).

The revised program will simulate this behaviour and so will correctly trace any program which falls foul of this bug, so assisting with the early detection of the problem.

In addition to the above enhancements, the revised program also includes the remedies for several bugs contained in the original program. In outline the bugs fixed are:

- The status register was not correctly updated when simulating the instruction PLA (See *Apple User* Feedback July 1986).
- The X register was not correctly updated when simulating the instruction TSX.
- The simulated stack was not used correctly when simulating the instructions JSR and RTS. One result is that any program which pushes an address into the stack and then executes an RTS to jump to that address will not be correctly traced by the original program. Since the same error was made consistently for both JSR and RTS, the original program correctly handles the more normal sequence of a JSR followed eventually by an RTS.
- Mistakenly entering an address with high byte = $C4 or $C6 or $C8 or $D4 as a break point address (command B) would also cause other commands of the original program to be executed.
- The original program fails to initialise the variables BREAKL and BREAKH. If these happen to contain an address in the program to be traced then you will never get past that address without manually resetting the break point address.

A final point on the use of the revised program. To determine where in memory you wish to assemble it you should be aware that the program requires $0613 bytes of contiguous memory. Listing I shows the program assembled at $0B00. From here to $0F13 is required by program code. A further $0100 bytes are then needed for the copy of page-0 and a further $0100 bytes for simulated stack. So from $0B00 the program requires memory up to and including $1113.

As I started by saying, I find this program extremely useful and I congratulate Michael Agerbak on his initial ideas for its implementation. I hope others find this revised version as useful as I do.

Command	Comments
Q	Quit. From the Tracer quit to the Editor, from the Editor quit the program.
G	Go. Available only in the Editor. Requires the input of a hex address. Activates the Tracer starting at the address specified.
B	Break point. Requires the input of a hex address. When the Tracer reaches the address specified it returns to the Editor without executing the instruction there. On returning to the Editor, the break point address is reset and the auto-repeat mode (command F) is cancelled.
A,X,Y,P	Change value of specified 6502 register. Requires the input of a single hex byte. The byte input is stored in the register specified.
C	Change memory location. Requires two lines of input, namely: (a) a hex address; followed by (b) a single hex byte. The byte input is stored in the memory location specified. Note, if the memory address lies in Page.0 or Page.1 then the byte will be stored in either the copy of Page.0 or in the simulated stack as appropriate.
D	Dump. Requires the input of a hex address. The contents of the specified memory location will be shown at the bottom of the output display. Note, if the memory address lies in Page.0 or Page.1 then the memory location displayed will be taken from the copy of Page.0 or the simulated stack as appropriate.
F	Fast. Sets Tracer to auto-repeat. Cancelled by (space), and also cancelled automatically on return to the Editor.
(space)	Single step. Available only in Tracer. Causes Tracer to execute the next instruction in the program being traced. Cancels auto-repeat.
H	Hi-res screen. Displays the HGR screen.
T	Text. Returns to the text display from HGR screen.

Table 1: Step by Step Version 2.0 commands

Listing I:

```
SOURCE FILE: STEP BY STEP V2.0 ($0B00)
                1 ****************************************
0000:           2 *
0000:           3 *     STEP BY STEP VERSION 2.0
0000:           4 *      A MACHINE CODE TRACER
0000:           5 *  BASED ON A PROGRAM PUBLISHED IN
0000:           6 *    THE APPLE USER MAGAZINE
0000:           7 *          MARCH 1986
0000:           8 *
0000:           9 *   REVISED BY TREVOR HOBSON
0000:          10 *       NOVEMBER 1986, AND
0000:          11 *       DECEMBER 1986
0000:          12 *
0000:          13 *
0000:          14 *
               15 ****************************************
0000:          16 *
0000:          17 *
0000:          18 *
0000:          19 *
0000:          20          DSECT
0000:          21 *
0000:          22 *
0000:          23 *
0000:          24 *
0000:          25 * ZERO PAGE LOCATIONS USED BY MONITOR ROUTINES
0024:          26 CH       EQU  $24
0025:          27 CV       EQU  $25
002F:          28 LENGTH   EQU  $2F
0033:          29 PROMPT   EQU  $33
003A:          30 PCL      EQU  $3A
003B:          31 PCH      EQU  $3B
003E:          32 A2L      EQU  $3E
003F:          33 A2H      EQU  $3F
0000:          34 *
0000:          35 *  OTHER ZERO PAGE LOCATIONS USED
0000:          36 P        EQU  $00        ;6502 REGISTERS
0001:          37 S        EQU  $01
0002:          38 AC       EQU  $02
0003:          39 X        EQU  $03
0004:          40 Y        EQU  $04
0000:          41 *
0009:          42 TEMP     EQU  $09
000A:          43 TEMP.P   EQU  $0A
000B:          44 OPCODE   EQU  $0B
000C:          45 LEN      EQU  $0C
000D:          46 ROMFLG   EQU  $0D
000E:          47 SPEED    EQU  $0E
000F:          48 D1L      EQU  $0F        ;DUMP BYTE
0010:          49 D1H      EQU  $10        ;ADDRESS
0011:          50 D2L      EQU  $11
0012:          51 D2H      EQU  $12
0013:          52 BREAKL   EQU  $13        ;BREAK POINT
0014:          53 BREAKH   EQU  $14
0015:          54 CHNGL    EQU  $15
0016:          55 CHNGH    EQU  $16
0017:          56 GETL     EQU  $17
0018:          57 GETH     EQU  $18
0019:          58 NXTPCL   EQU  $19
001A:          59 NXTPCH   EQU  $1A
001B:          60 THSPCL   EQU  $1B
001C:          61 THSPCH   EQU  $1C
0000:          62 *
0000:          63 *
0000:          64 *
0000:          65 *       CONSTANTS
00CF:          66 ROMH     EQU  $CF
0100:          67 PGLEN    EQU  $0100
0000:          68 *
0000:          69 *
0000:          70 *      DOS LOCATION
03D0:          71 DOS      EQU  $03D0
0000:          72 *
0000:          73 *
0000:          74 *   MONITOR ROUTINES USED
F94B:          75 PRBLNK   EQU  $F94B
FBDD:          76 BELL1    EQU  $FBDD
FC42:          77 CLREOP   EQU  $FC42
FC58:          78 HOME     EQU  $FC58
FD35:          79 RDCHAR   EQU  $FD35
FD67:          80 GETLNZ   EQU  $FD67
FDBE:          81 CROUT    EQU  $FDBE
FDDA:          82 PRBYTE   EQU  $FDDA
FDE3:          83 PRHEX    EQU  $FDE3
FDED:          84 COUT     EQU  $FDED
FE63:          85 LIST2    EQU  $FE63
FE80:          86 SETINV   EQU  $FE80
FE84:          87 SETNORM  EQU  $FE84
FFA7:          88 GETNUM   EQU  $FFA7
0000:          89 *
0000:          90 *
0000:          91 *
0000:          92          DEND
0000:          93 *
0000:          94 *
0000:          95 *
0000:          96 *
----- NEXT OBJECT FILE NAME IS STEP BY STEP V2.0 ($0B00).OBJ0
0B00:          97          ORG  $0B00
0B00:          98 *
0B00:          99 *
0B00:         100 *
0B00:         101 *
              102 ****************************************
0B00:         103 * START
0B00:20 2D 0E 104          JSR  SAVEZP     ;SAVE ZERO PAGE
0B03:A2 00    105          LDX  #$00       ;SET UP
0B05:86 00    106          STX  P          ;STATUS REGISTER
0B07:86 0D    107          STX  ROMFLG     ;ROM FLAG
0B09:CA       108          DEX
0B0A:86 01    109          STX  S          ;STACK POINTER
0B0C:A9 BA    110          LDA  #':        ;PROMPT
0B0E:85 33    111          STA  PROMPT
              112 ****************************************
0B10:A2 00    113 EDITOR   LDX  #$00       ;CANCEL AUTO-REPEAT
0B12:86 0E    114          STX  SPEED
0B14:CA       115          DEX
0B15:86 14    116          STX  BREAKH     ;RE-SET BREAKPOINT ADDRESS
0B17:86 13    117          STX  BREAKL
0B19:20 58 FC 118          JSR  HOME       ;CLEAR SCREEN
0B1C:20 80 FE 119          JSR  SETINV     ;SET INVERSE
0B1F:20 73 0D 120          JSR  DISPLY     ;DISPLAY
0B22:20 84 FE 121          JSR  SETNORM    ;SET NORMAL
0B25:20 35 FD 122          JSR  RDCHAR     ;GET INPUT
0B28:20 ED FD 123          JSR  COUT       ;OUTPUT CHARACTER
0B2B:C9 D1    124          CMP  #'Q        ;QUIT?
0B2D:D0 09    125          BNE  NOQ
0B2F:20 38 0E 126          JSR  FLIPZP     ;RESTORE ZERO PAGE
0B32:20 58 FC 127          JSR  HOME
0B35:4C D0 03 128          JMP  DOS        ;EXIT TO DOS
              129 ****************************************
0B38:C9 C7    130 NOQ      CMP  #'G        ;INPUT = G?
0B3A:D0 13    131          BNE  NOG
0B3C:20 67 FD 132          JSR  GETLNZ     ;GET ADDRESS
0B3F:A0 00    133          LDY  #$00       ;AND
0B41:20 A7 FF 134          JSR  GETNUM
0B44:A5 3E    135          LDA  A2L
0B46:85 19    136          STA  NXTPCL     ;PUT IN NEXT PC
0B48:A5 3F    137          LDA  A2H
0B4A:85 1A    138          STA  NXTPCH
0B4C:4C 55 0B 139          JMP  TRACE      ;THEN TRACE
0B4F:20 4A 0E 140 NOG      JSR  EDKEY1
0B52:4C 10 0B 141          JMP  EDITOR     ;THEN RETURN
              142 ****************************************
0B55:A5 19    143 TRACE    LDA  NXTPCL     ;GET NEXT PC
0B57:85 1B    144          STA  THSPCL     ;AND PUT INTO PC
0B59:C5 13    145          CMP  BREAKL     ;IF BREAK POINT
0B5B:D0 09    146          BNE  NOTBK1
0B5D:A5 1A    147          LDA  NXTPCH
0B5F:C5 14    148          CMP  BREAKH
0B61:D0 05    149          BNE  NOTBK2
0B63:4C 10 0B 150          JMP  EDITOR     ;THEN JUMP TO EDITOR
0B66:A5 1A    151 NOTBK1   LDA  NXTPCH     ;ELSE CONTINUE
0B68:85 1C    152 NOTBK2   STA  THSPCH
0B6A:20 73 0D 153 LOOP1    JSR  DISPLY     ;DISPLAY
0B6D:24 0E    154          BIT  SPEED      ;TEST TRACE SPEED
0B6F:50 0B    155          BVC  SKP1       ;IF NOT FAST THEN JUMP ON
0B71:AD 00 C0 156          LDA  $C000      ;READ LAST KEY PRESSED
0B74:C9 A0    157          CMP  #$A0       ;IF SPACE THEN
0B76:D0 1B    158          BNE  CONTIN
0B78:A9 00    159          LDA  #$00       ;SLOW MODELAB
0B7A:85 0E    160          STA  SPEED
0B7C:20 35 FD 161 SKP1     JSR  RDCHAR     ;AND GET INPUT
0B7F:20 ED FD 162          JSR  COUT       ;AND OUTPUT
0B82:C9 D1    163          CMP  #'Q        ;IF = Q THEN
0B84:D0 03    164          BNE  SKP2
0B86:4C 10 0B 165          JMP  EDITOR     ;JUMP TO EDITOR
0B89:C9 A0    166 SKP2     CMP  #$A0       ;IF = SPACE
0B8B:F0 06    167          BEQ  CONTIN     ;CONTINUE TRACE
0B8D:20 4A 0E 168          JSR  EDKEY1     ;IF = ANYTHING ELSE
0B90:4C 6A 0B 169          JMP  LOOP1      ;THEN GO BACK ABOVE
0B93:A5 1C    170 CONTIN   LDA  THSPCH     ;CONTINUE...
0B95:C9 CF    171          CMP  #ROMH      ;TEST TO SEE IF
0B97:90 29    172          BCC  NOTROM     ;INSTRUCTION IS IN ROM
0B99:20 BE FD 173          JSR  CROUT      ;IF NOT THEN SKIP QUERY
0B9C:A2 00    174          LDX  #$00
0B9E:BD 0B 0F 175 LOOP2    LDA  QUERY,X
0BA1:20 ED FD 176          JSR  COUT
0BA4:E8       177          INX
0BA5:C9 BF    178          CMP  #'?        ;AND
0BA7:D0 F5    179          BNE  LOOP2
0BA9:20 35 FD 180 ANS      JSR  RDCHAR     ;GET ANSWER
0BAC:C9 CE    181          CMP  #'N        ;IF NO
0BAE:F0 0F    182          BEQ  NODOROM
0BB0:C9 D9    183          CMP  #'Y
0BB2:F0 06    184          BEQ  DOROM
0BB4:20 DD FB 185          JSR  BELL1
0BB7:4C A9 0B 186          JMP  ANS
0BBA:C6 0D    187 DOROM    DEC  ROMFLG     ;SET ROM FLAG
0BBC:4C 31 0D 188          JMP  DO.IT
0BBF:4C 39 0C 189 NODOROM  JMP  DO.RTS
0BC2:20 58 FC 190 NOTROM   JSR  HOME
0BC5:20 19 0E 191          JSR  LINE       ;DISASSEMBLE LINE
0BC8:18       192          CLC
0BC9:65 1B    193          ADC  THSPCL
0BCB:85 19    194          STA  NXTPCL     ;AND WORK OUT
0BCD:A5 1C    195          LDA  THSPCH     ;NEXT PCL BY ADDING
0BCF:69 00    196          ADC  #$00       ;LENGTH TO PC
0BD1:85 1A    197          STA  NXTPCH
0BD3:A4 0C    198          LDY  LEN        ;THEN GET NEXT INSTRUCTION
0BD5:B1 1B    199          LDA  (THSPCL),Y ;AND SAVE IT
0BD7:85 0B    200          STA  OPCODE
0BD9:A9 60    201          LDA  #$60       ;PUTTING AN RTS
0BDB:91 1B    202          STA  (THSPCL),Y ;WHERE IT WAS
0BDD:A0 00    203          LDY  #$00       ;THEN GET THIS
0BDF:B1 1B    204          LDA  (THSPCL),Y ;INSTRUCTION AND TEST
              205 ****************************************
0BE1:C9 00    206          CMP  #$00       ;BRK
0BE3:D0 09    207          BNE  JSR
0BE5:A4 0C    208 RTN      LDY  LEN        ;IF BRK THEN
0BE7:A5 0B    209          LDA  OPCODE     ;RESTORE NEXT INSTRUCTION
0BE9:91 1B    210          STA  (THSPCL),Y ;AND RETURN
0BEB:4C 10 0B 211          JMP  EDITOR     ;TO EDITOR
              212 ****************************************
0BEE:C9 20    213 JSR      CMP  #$20       ;IS THE INSTRUCTION A JSR?
0BF0:D0 26    214          BNE  RTI
0BF2:38       215          SEC
0BF3:A5 19    216          LDA  NXTPCL     ;DECREMENT NXTPC
0BF5:E9 01    217          SBC  #$01
0BF7:85 19    218          STA  NXTPCL
0BF9:A5 1A    219          LDA  NXTPCH
0BFB:E9 00    220          SBC  #$00
0BFD:A6 01    221          LDX  S          ;PUT NXTPC IN STACK
0BFF:9D 14 10 222          STA  PAGE.0+PGLEN,X
0C02:CA       223          DEX
0C03:A5 19    224          LDA  NXTPCL
0C05:9D 14 10 225          STA  PAGE.0+PGLEN,X
0C08:CA       226          DEX
0C09:86 01    227          STX  S
0C0B:C8       228          INY
0C0C:B1 1B    229          LDA  (THSPCL),Y
0C0E:85 19    230          STA  NXTPCL
0C10:C8       231          INY
0C11:B1 1B    232          LDA  (THSPCL),Y
0C13:85 1A    233          STA  NXTPCH
0C15:4C 59 0D 234          JMP  RESTOR
              235 ****************************************
0C18:C9 40    236 RTI      CMP  #$40       ;IS IT AN RTI?
0C1A:D0 19    237          BNE  RTS
0C1C:A6 01    238          LDX  S
0C1E:E8       239          INX
0C1F:BD 14 10 240          LDA  PAGE.0+PGLEN,X
0C22:85 00    241          STA  P
0C24:E8       242          INX
```

DISK MANAGEMENT SYSTEMS
The Pocket Pak will hold up to 10 3.5" Microdisks and is small enough to fit inside a coat, jacket or purse. Ideal for students, professors or executives on the move. Available in nylon with a variety of colours. **£9.95 + VAT**
The Easel as above but holds up to 20 3.5" Microdisks and folds to a thickness of only 1". Now shipping new improved model **£12.95 + VAT**
The Pyramid holds up to 24 5.25" floppy disks with a storage design that allows the user to quickly read each disk label at a glance. Designed for the executive, professional or student who is on the move. Available in a variety of colours at **£12.95 + VAT**

Apple IIc Owners!

With **Powerwedge** battery back-up you can use your computer anywhere. 2 Hour operation from one charge - 3 Hour recharge from IIc mains unit. Automatic shut-off - Fused output - LED Charge indicator -Light Weight. As used by Chris Bonnington during his conquest of Everest.

£89.00 plus VAT
Further details on request

NEW LOWER PRICE FOR APPLE IIGS

Yes in response to many customers wishing to purchase the computer only, we are now able to offer the Apple IIGS 256k for £650 + VAT

Apple IIGS 256K	**£650.00**
Apple Drive 3.5	£250.00
Apple Drive 5.25	£175.00
Apple Hard Disc 20SC	£895.00
Apple Hard Disc 40SC	£1295.00
Apple Mono Monitor	£90.00
Apple RGB Colour Monitor	£375.00
Macintosh SE Keyboard for IIGS	*£99.00*
Apple SCSI Interface Card	£65.00
Apple SCSI Peripheral Cable	£35.00
Apple Peripheral Adaptor Cable	£20.00
Apple Exp Card 256k	£95.00
Apple Extra memory per 256K bank	£25.00
Apple Exp Card 1 Megabyte	£175.00
Apple Fan Kit	£50.00
Apple Joystick	£39.00
IIGS Monochrome System 256k	*£895.00*
IIGS Monochrome System 1Mb	*£995.00*
IIGS Colour System 512k	*£1195.00*
IIGS Hard Disk System 512k	*£1895.00*
IIGS Hard Disk Colour System	*£2150.00*

For Immediate Part Exchange Quotations ring HOLDENS APPLEXCHANGE 0772 561321

HOLDENS COMPUTER SUPPLIES

for new and used

Apple & Macintosh Computers

also the fantastic

Apple Desktop Publishing System

191-195 Marsh Lane, Preston PR1 8NL. Tel 0772 52686 or 561321

MouseStuff

This new utility program has really taken off and is proving extremely popular with Pascal programmers all over the world

For Apple IIc or IIe with 80 col.
Written in assembler and allowing Pascal programmers to use pull-down menus in their own programs. These utilities also allow saving or clearing, part or all of the screen and also incorporate a speedy routine for writing to the screen. Full use of the mouse is possible, but menus may also be opened from the keyboard. Assembler knowledge not necessary.

£39.00 + VAT
Fact sheet available or read-
Max Parrot's excellent revue in last month's Apple User!

NEW for OLD!

An excellent opportunity to considerably enhance your system. Change to the latest Imagewriter II and add speed and colour to your life. We will allow £150 for your ImageWriter I against the purchase of this superb new printer.

Imagewriter II £349.00
Peripheral 8 Cable £25.00
(Plus VAT)

AppleCare
A new extended warranty scheme enabling Apple users to extend their normal warranty period for a further 12 months at a cost lower than the average repair charge.
Listed below are typical systems and costs.
Quotations for other configurations can be arranged.

Apple II or IIe, DuoDisk or 2 Disk Drives, Monitor II or III & Imagewriter	**£170.00**
Apple IIc, External Drive, Monitor IIc & Imagewriter	**£175.00**
Apple IIGS, External Drive, Monitor & Imagewriter	**£150.00**
Macintosh	**£150.00**
Macintosh, External Drive & Imagewriter	**£225.00**
Macintosh, Hard Disk 20 & Imagewriter	**£365.00**
Macintosh, Hard Disk 20 & LaserWriter	**£665.00**
LaserWriter	**£350.00**

Please add VAT to all prices quoted

```
0C25:BD 14 10  243          LDA  PAGE.0+PGLEN,X
0C28:85 19     244          STA  NXTPCL
0C2A:E8        245          INX
0C2B:BD 14 10  246          LDA  PAGE.0+PGLEN,X
0C2E:85 1A     247          STA  NXTPCH
0C30:86 01     248          STX  S
0C32:4C 59 0D  249          JMP  RESTOR
               250 ****************************************
0C35:C9 60     251 RTS      CMP  #$60       ;IS IT AN RTS?
0C37:D0 1F     252          BNE  JMP1
0C39:A6 01     253 DO.RTS   LDX  S          ;IF SO THEN CHECK POINTER
0C3B:E0 FF     254          CPX  #$FF       ;IS NOT AT BEGINNING
0C3D:D0 03     255          BNE  NEFF
0C3F:4C E5 0B  256          JMP  RTN        ;IF IT IS THEN JUMP TO EDITOR
0C42:E8        257 NEFF     INX             ;IF NOT
0C43:BD 14 10  258          LDA  PAGE.0+PGLEN,X
0C46:18        259          CLC
0C47:69 01     260          ADC  #$01
0C49:85 19     261          STA  NXTPCL
0C4B:E8        262          INX
0C4C:BD 14 10  263          LDA  PAGE.0+PGLEN,X
0C4F:69 00     264          ADC  #$00
0C51:85 1A     265          STA  NXTPCH
0C53:86 01     266          STX  S
0C55:4C 59 0D  267          JMP  RESTOR
               268 ****************************************
0C58:C9 4C     269 JMP1     CMP  #$4C       ;IS IT A JMP?
0C5A:D0 0D     270          BNE  JMP2
0C5C:C8        271          INY             ;IF SO GET OPERAND
0C5D:B1 1B     272          LDA  (THSPCL),Y ;AND PUT IN
0C5F:85 19     273          STA  NXTPCL     ;NXTPCL
0C61:C8        274          INY
0C62:B1 1B     275          LDA  (THSPCL),Y
0C64:85 1A     276          STA  NXTPCH
0C66:4C 59 0D  277          JMP  RESTOR
               278 ****************************************
0C69:C9 6C     279 JMP2     CMP  #$6C       ;IS IT A JMP ()?
0C6B:D0 27     280          BNE  BRNCH
0C6D:C8        281          INY             ;IF SO GET OPERAND
0C6E:B1 1B     282          LDA  (THSPCL),Y ;AND THEN
0C70:85 17     283          STA  GETL       ;LOAD VALUE
0C72:C8        284          INY             ;AT OPERAND ADDRESS
0C73:B1 1B     285          LDA  (THSPCL),Y ;GET OPERAND
0C75:C9 02     286          CMP  #$02
0C77:B0 0A     287          BCS  NOTTMP     ;IF IN PAGE.0 OR PAGE.1
0C79:A9 14     288          LDA  #>PAGE.0   ;THEN USE THE TEMPORARY
0C7B:65 17     289          ADC  GETL       ;STORAGE AREA
0C7D:85 17     290          STA  GETL
0C7F:A9 0F     291          LDA  #<PAGE.0
0C81:71 1B     292          ADC  (THSPCL),Y
0C83:85 18     293 NOTTMP   STA  GETH
0C85:A0 00     294          LDY  #$00
0C87:B1 17     295          LDA  (GETL),Y   ;IN NEXTPC
0C89:85 19     296          STA  NXTPCL
0C8B:E6 17     297          INC  GETL
0C8D:          298 * NB: THE ABOVE INSTRUCTION SHOULD REALLY BE
0C8D:          299 *               INY
0C8D:          300 *     HOWEVER AS IT NOW STANDS IT WILL CORRECTLY
0C8D:          301 *     SIMULATE THE FOLLOWING BUG IN THE 6502.
0C8D:          302 *     WHENEVER THE 6502 ENCOUNTERS THE INSTRUCTION
0C8D:          303 *     JMP(XXFF) (WHERE XX IS ANY VALID PAGE NUMBER)
0C8D:          304 *     A JUMP WILL TAKE PLACE TO AN ADDRESS GIVEN BY
0C8D:          305 *          LOW BYTE FROM XXFF
0C8D:          306 *        HIGH BYTE FROM XX00  (SHOULD BE (XX+1)00)
0C8D:B1 17     307          LDA  (GETL),Y
0C8F:85 1A     308          STA  NXTPCH
0C91:4C 59 0D  309          JMP  RESTOR
               310 ****************************************
0C94:48        311 BRNCH    PHA             ;IS IT A BRANCH?
0C95:29 1F     312          AND  #$1F       ;IF IT IS IT MUST BE
0C97:C9 10     313          CMP  #$10       ;XXX10000
0C99:D0 2F     314          BNE  STCK       ;NOT A BRANCH
0C9B:68        315          PLA             ;IS A BRANCH SO RESTORE
0C9C:8D A4 0C  316          STA  DOBRN      ;AND EXECUTE
0C9F:A5 00     317          LDA  P          ;THE BRANCH AFTER
0CA1:48        318          PHA             ;LOADING IN
0CA2:28        319          PLP             ;THE STATUS REGISTER
0CA3:D8        320          CLD             ;NECESSARY TO PREVENT CORRUPTION
0CA4:F0 03     321 DOBRN    BEQ  POS        ;IF NOT BRANCH CONDITION
0CA6:4C 59 0D  322          JMP  RESTOR     ;THEN RESTOR
0CA9:C8        323 POS      INY             ;IF BRANCH CONDITION
0CAA:B1 1B     324          LDA  (THSPCL),Y ;THEN DO RELATIVE BRANCH
0CAC:30 0E     325          BMI  SUB        ;BY SUBTRACTING OR
0CAE:18        326          CLC             ;ADDING TO PC
0CAF:65 19     327          ADC  NXTPCL
0CB1:85 19     328          STA  NXTPCL
0CB3:A5 1A     329          LDA  NXTPCH
0CB5:69 00     330          ADC  #$00
0CB7:85 1A     331          STA  NXTPCH
0CB9:4C 59 0D  332          JMP  RESTOR
0CBC:18        333 SUB      CLC
0CBD:65 19     334          ADC  NXTPCL
0CBF:85 19     335          STA  NXTPCL
0CC1:A5 1A     336          LDA  NXTPCH
0CC3:69 FF     337          ADC  #$FF
0CC5:85 1A     338          STA  NXTPCH
0CC7:4C 59 0D  339          JMP  RESTOR
               340 ****************************************
0CCA:68        341 STCK     PLA             ;STACK OPERATIONS
0CCB:C9 9A     342          CMP  #$9A       ;TXS
0CCD:D0 07     343          BNE  PHA
0CCF:A6 03     344          LDX  X
0CD1:86 01     345          STX  S
0CD3:4C 59 0D  346          JMP  RESTOR
0CD6:C9 48     347 PHA      CMP  #$48       ;PHA
0CD8:D0 0C     348          BNE  PLA
0CDA:A6 01     349          LDX  S
0CDC:A5 02     350          LDA  AC
0CDE:9D 14 10  351          STA  PAGE.0+PGLEN,X
0CE1:C6 01     352          DEC  S
0CE3:4C 59 0D  353          JMP  RESTOR
0CE6:C9 68     354 PLA      CMP  #$68       ;PLA
0CE8:D0 14     355          BNE  PHP
0CEA:E6 01     356          INC  S
0CEC:A6 01     357          LDX  S
0CEE:A5 00     358          LDA  P
0CF0:48        359          PHA
0CF1:28        360          PLP
0CF2:BD 14 10  361          LDA  PAGE.0+PGLEN,X
0CF5:85 02     362          STA  AC
0CF7:08        363          PHP
0CF8:68        364          PLA
0CF9:85 00     365          STA  P
0CFB:4C 59 0D  366          JMP  RESTOR
0CFE:C9 08     367 PHP      CMP  #$08       ;PHP
0D00:D0 0C     368          BNE  PLP
0D02:A6 01     369          LDX  S
0D04:A5 00     370          LDA  P
0D06:9D 14 10  371          STA  PAGE.0+PGLEN,X
0D09:C6 01     372          DEC  S
0D0B:4C 59 0D  373          JMP  RESTOR
0D0E:C9 28     374 PLP      CMP  #$28       ;PLP
0D10:D0 0C     375          BNE  TSX
0D12:E6 01     376          INC  S
0D14:A6 01     377          LDX  S
0D16:BD 14 10  378          LDA  PAGE.0+PGLEN,X
0D19:85 00     379          STA  P
0D1B:4C 59 0D  380          JMP  RESTOR
0D1E:C9 BA     381 TSX      CMP  #$BA       ;TSX
0D20:D0 0F     382          BNE  DO.IT
0D22:A5 00     383          LDA  P
0D24:48        384          PHA
0D25:28        385          PLP
0D26:A5 01     386          LDA  S
0D28:85 03     387          STA  X
0D2A:08        388          PHP
0D2B:68        389          PLA
0D2C:85 00     390          STA  P
0D2E:4C 59 0D  391          JMP  RESTOR
               392 ****************************************
0D31:08        393 DO.IT    PHP             ;SAVE P
0D32:68        394          PLA
0D33:85 0A     395          STA  TEMP.P
0D35:20 38 0E  396          JSR  FLIPZP     ;FLIP ZERO PAGE
0D38:20 62 0D  397          JSR  GO         ;THEN EXECUTE INSTRUCTION
0D3B:08        398          PHP             ;SAVE ALL REGISTERS
0D3C:8D 16 0F  399          STA  PAGE.0+AC
0D3F:68        400          PLA             ;IN ORDER NOT TO
0D40:8D 14 0F  401          STA  PAGE.0+P
0D43:8E 17 0F  402          STX  PAGE.0+X   ;DISTURB
0D46:8C 18 0F  403          STY  PAGE.0+Y
0D49:20 38 0E  404          JSR  FLIPZP     ;FLIP ZERO PAGE
0D4C:A5 0A     405          LDA  TEMP.P
0D4E:48        406          PHA
0D4F:28        407          PLP
0D50:24 0D     408          BIT  ROMFLG     ;CHECK IF JUST EXECUTED
0D52:10 05     409          BPL  RESTOR     ;A ROM ROUTINE
0D54:E6 0D     410          INC  ROMFLG     ;IF SO THEN CLEAR FLAG
0D56:4C 39 0C  411          JMP  DO.RTS     ;AND FORCE AN RTS
               412 ****************************************
0D59:A4 0C     413 RESTOR   LDY  LEN        ;RESTORE
0D5B:A5 0B     414          LDA  OPCODE     ;INSTRUCTION
0D5D:91 1B     415          STA  (THSPCL),Y
0D5F:4C 55 0B  416          JMP  TRACE      ;GO BACK AND DO IT ALL AGAIN
               417 ****************************************
0D62:AC 18 0F  418 GO       LDY  PAGE.0+Y
0D65:AE 17 0F  419          LDX  PAGE.0+X
0D68:AD 14 0F  420          LDA  PAGE.0+P
0D6B:48        421          PHA
0D6C:AD 16 0F  422          LDA  PAGE.0+AC
0D6F:28        423          PLP
0D70:6C 2F 0F  424          JMP  (PAGE.0+THSPCL)
               425 ****************************************
0D73:          426 * NB: DUE TO A BUG IN THE 6502 PROCESSOR
0D73:          427 *     A JMP($XXFF) WILL NOT WORK AS EXPECTED.
0D73:          428 *     AFTER MAKING ANY MODIFICATIONS TO THIS
0D73:          429 *     PROGRAM, CHECK THAT THE ABOVE INSTRUCTION
0D73:          430 *          JMP (PAGE.0+THSPCL)
0D73:          431 *     DOES NOT HAVE A LOW BYTE OF $FF.
0D73:          432 *     IF IT DOES THEN CHANGE THE PAGE 0
0D73:          433 *     LOCATION OF:
0D73:          434 *          THSPCL & THSPCH
0D73:          435 *     AND THEN RE-ASSEMBLE.
               436 ****************************************
0D73:A9 00     437 DISPLY   LDA  #$00
0D75:85 24     438          STA  CH
0D77:A9 02     439          LDA  #$02
0D79:85 25     440          STA  CV
0D7B:20 42 FC  441          JSR  CLREOP
0D7E:20 19 0E  442          JSR  LINE       ;DISASSEMBLE NEXT LINE
0D81:A9 05     443          LDA  #$05
0D83:20 63 FE  444          JSR  LIST2      ;DISASSEMBLE 5 MORE LINES
0D86:20 8E FD  445          JSR  CROUT
0D89:20 8E FD  446          JSR  CROUT
0D8C:A0 0D     447          LDY  #$0D
0D8E:B9 E5 0D  448 FLAGS    LDA  DISP1,Y
0D91:20 ED FD  449          JSR  COUT
0D94:88        450          DEY
0D95:D0 F7     451          BNE  FLAGS
0D97:20 8E FD  452 NEXTR    JSR  CROUT
0D9A:20 F3 0D  453          JSR  PRNTIT
0D9D:C8        454          INY
0D9E:C0 05     455          CPY  #$05
0DA0:D0 F5     456          BNE  NEXTR
0DA2:20 8E FD  457          JSR  CROUT
0DA5:20 8E FD  458          JSR  CROUT
0DA8:A9 A8     459          LDA  #'(        ;DUMP
0DAA:20 ED FD  460          JSR  COUT
0DAD:A5 10     461          LDA  D1H
0DAF:20 DA FD  462          JSR  PRBYTE
0DB2:A5 0F     463          LDA  D1L
0DB4:85 11     464          STA  D2L
0DB6:20 DA FD  465          JSR  PRBYTE
0DB9:A9 A9     466          LDA  #')
0DBB:20 ED FD  467          JSR  COUT
0DBE:A9 BA     468          LDA  #':
0DC0:20 ED FD  469          JSR  COUT
0DC3:A5 10     470          LDA  D1H
0DC5:C9 02     471          CMP  #$02       ;IS IT IN PAGE.0
0DC7:B0 0A     472          BCS  DFIN       ;OR THE STACK?
0DC9:A9 14     473          LDA  #>PAGE.0
0DCB:65 11     474          ADC  D2L
0DCD:85 11     475          STA  D2L
0DCF:A9 0F     476          LDA  #<PAGE.0
0DD1:65 10     477          ADC  D1H
0DD3:85 12     478 DFIN     STA  D2H
0DD5:A2 00     479          LDX  #$00
0DD7:A1 11     480          LDA  (D2L,X)
0DD9:20 DA FD  481          JSR  PRBYTE
0DDC:20 8E FD  482          JSR  CROUT
0DDF:60        483          RTS
0DE0:D0 D3 C1  484 DISP     ASC  'PSAXY'
0DE3:DB D9
0DE5:A0 A0     485 DISP1    DFB  $A0,$A0
```

```
0DE7:C3 DA C9  486          ASC  'CZIDB-VN
0DEA:C4 C2 AD
0DED:D6 CE A0
0DF0:A0 A0 A0
0DF3:B9 E0 0D  487 PRNTIT   LDA  DISP,Y
0DF6:20 ED FD  488          JSR  COUT
0DF9:20 48 F9  489          JSR  PRBLNK
0DFC:B9 00 00  490          LDA  P,Y
0DFF:85 09     491          STA  TEMP
0E01:A2 08     492          LDX  #$08
0E03:06 09     493 SHIFT    ASL  TEMP
0E05:A9 00     494          LDA  #$00
0E07:69 00     495          ADC  #$00
0E09:20 E3 FD  496          JSR  PRHEX
0E0C:CA        497          DEX
0E0D:D0 F4     498          BNE  SHIFT
0E0F:20 48 F9  499          JSR  PRBLNK
0E12:B9 00 00  500          LDA  P,Y
0E15:20 DA FD  501          JSR  PRBYTE
0E18:60        502          RTS
               503 ************************************
0E19:A5 1B     504 LINE     LDA  THSPCL
0E1B:85 3A     505          STA  PCL
0E1D:A5 1C     506          LDA  THSPCH
0E1F:85 3B     507          STA  PCH
0E21:A9 01     508          LDA  #$01
0E23:20 63 FE  509          JSR  LIST2
0E26:A4 2F     510          LDY  LENGTH
0E28:C8        511          INY
0E29:84 0C     512          STY  LEN
0E2B:98        513          TYA
0E2C:60        514          RTS
               515 ************************************
0E2D:A2 00     516 SAVEZP   LDX  #$00       ;SAVE ZERO PAGE
0E2F:B5 00     517 LOOP3    LDA  $00,X
0E31:9D 14 0F  518          STA  PAGE.0,X
0E34:CA        519          DEX
0E35:D0 F8     520          BNE  LOOP3
0E37:60        521          RTS
               522 ************************************
0E38:A2 00     523 FLIPZP   LDX  #$00       ;FLIP ZERO PAGE
0E3A:B5 00     524 LOOP4    LDA  $00,X
0E3C:48        525          PHA
0E3D:BD 14 0F  526          LDA  PAGE.0,X
0E40:95 00     527          STA  $00,X
0E42:68        528          PLA
0E43:9D 14 0F  529          STA  PAGE.0,X
0E46:CA        530          DEX
0E47:D0 F1     531          BNE  LOOP4
0E49:60        532          RTS
               533 ************************************
0E4A:C9 C1     534 EDKEY1   CMP  #'A        ;INPUT = A?
0E4C:D0 0D     535          BNE  EQX
0E4E:20 67 FD  536          JSR  GETLNZ
0E51:A0 00     537          LDY  #$00
0E53:20 A7 FF  538          JSR  GETNUM
0E56:A5 3E     539          LDA  A2L
0E58:85 02     540          STA  AC
0E5A:60        541          RTS
0E5B:C9 D8     542 EQX      CMP  #'X        ;INPUT = X?
0E5D:D0 0D     543          BNE  EQP
0E5F:20 67 FD  544          JSR  GETLNZ
0E62:A0 00     545          LDY  #$00
0E64:20 A7 FF  546          JSR  GETNUM
0E67:A5 3E     547          LDA  A2L
0E69:85 03     548          STA  X
0E6B:60        549          RTS
0E6C:C9 D0     550 EQP      CMP  #'P        ;INPUT = P?
0E6E:D0 0D     551          BNE  EQY
0E70:20 67 FD  552          JSR  GETLNZ
0E73:A0 00     553          LDY  #$00
0E75:20 A7 FF  554          JSR  GETNUM
0E78:A5 3E     555          LDA  A2L
0E7A:85 00     556          STA  P
0E7C:60        557          RTS
0E7D:C9 D9     558 EQY      CMP  #'Y        ;INPUT = Y?
0E7F:D0 0D     559          BNE  EQB
0EB1:20 67 FD  560          JSR  GETLNZ
0EB4:A0 00     561          LDY  #$00
0EB6:20 A7 FF  562          JSR  GETNUM
0EB9:A5 3E     563          LDA  A2L
0EBB:85 04     564          STA  Y
0EBD:60        565          RTS
0EBE:C9 C2     566 EQB      CMP  #'B        ;INPUT = B?
0E90:D0 11     567          BNE  EQH
0E92:20 67 FD  568          JSR  GETLNZ
0E95:A0 00     569          LDY  #$00
0E97:20 A7 FF  570          JSR  GETNUM
0E9A:A5 3E     571          LDA  A2L
0E9C:85 13     572          STA  BREAKL
0E9E:A5 3F     573          LDA  A2H
0EA0:85 14     574          STA  BREAKH
0EA2:60        575          RTS
0EA3:C9 CB     576 EQH      CMP  #'H        ;INPUT = H?
0EA5:D0 0A     577          BNE  EQT
0EA7:AD 50 C0  578          LDA  $C050
0EAA:AD 53 C0  579          LDA  $C053
0EAD:AD 57 C0  580          LDA  $C057
0EB0:60        581          RTS
0EB1:C9 D4     582 EQT      CMP  #'T        ;INPUT = T?
0EB3:D0 07     583          BNE  EQD
0EB5:AD 51 C0  584          LDA  $C051
0EB8:AD 54 C0  585          LDA  $C054
0EBB:60        586          RTS
0EBC:C9 C4     587 EQD      CMP  #'D        ;INPUT = D?
0EBE:D0 11     588          BNE  EQF
0EC0:20 67 FD  589          JSR  GETLNZ
0EC3:A0 00     590          LDY  #$00
0EC5:20 A7 FF  591          JSR  GETNUM
0EC8:A5 3E     592          LDA  A2L
0ECA:85 0F     593          STA  D1L
0ECC:A5 3F     594          LDA  A2H
0ECE:85 10     595          STA  D1H
0ED0:60        596          RTS
0ED1:C9 C6     597 EQF      CMP  #'F        ;INPUT = F?
0ED3:D0 05     598          BNE  EQC
0ED5:A0 FF     599          LDY  #$FF
0ED7:84 0E     600          STY  SPEED
0ED9:60        601          RTS
0EDA:C9 C3     602 EQC      CMP  #'C        ;INPUT = C?
0EDC:D0 2C     603          BNE  END
0EDE:20 67 FD  604          JSR  GETLNZ
0EE1:A0 00     605          LDY  #$00
0EE3:20 A7 FF  606          JSR  GETNUM
0EE6:A5 3E     607          LDA  A2L
0EE8:85 15     608          STA  CHNGL
0EEA:A5 3F     609          LDA  A2H
0EEC:C9 02     610          CMP  #$02       ;IS IT IN PAGE.0
0EEE:B0 0A     611          BCS  SKPC       ;OR THE STACK?
0EF0:A9 14     612          LDA  #>PAGE.0
0EF2:65 15     613          ADC  CHNGL
0EF4:85 15     614          STA  CHNGL
0EF6:A9 0F     615          LDA  #<PAGE.0
0EF8:65 3F     616          ADC  A2H
0EFA:85 16     617 SKPC     STA  CHNGH
0EFC:20 67 FD  618          JSR  GETLNZ
0EFF:A0 00     619          LDY  #$00
0F01:20 A7 FF  620          JSR  GETNUM
0F04:A5 3E     621          LDA  A2L
0F06:A0 00     622          LDY  #$00
0F08:91 15     623          STA  (CHNGL),Y
0F0A:60        624 END      RTS
               625 ************************************
0F0B:A0 C5 D8  626 QUERY    ASC  '          EXECUTE?'
0F0E:C5 C3 D5
0F11:D4 C5 BF
               627 ************************************
0F14:          628 PAGE.0   DS   1
0F15:          629 *    THE TEMPORARY PAGE.0 STORAGE AREA REQUIRES
0F15:          630 *    $0100 BYTES STARTING AT THE ADDRESS OF THE
0F15:          631 *    LABEL 'PAGE.0'.
0F15:          632 *    THE SIMULATED STACK STORAGE AREA THEN
0F15:          633 *    REQUIRES A FURTHER $0100 BYTES FROM THE
0F15:          634 *    END OF THE TEMPORARY PAGE.0 STORAGE AREA.
```

Listing II

```
SOURCE FILE: TEST STEP BY STEP V2.0
               1 ************************************
0000:          2 *
0000:          3 *   TEST FOR STEP BY STEP V2.0
0000:          4 *
               5 ************************************
0000:          6 *
0000:          7          DSECT
000C:          8 CA       EQU  $0C
000D:          9 CB       EQU  $0D
000E:         10 CC       EQU  $0E
000F:         11 CD       EQU  $0F
0010:         12 CE       EQU  $10
0000:         13 *
FC58:         14 HOME     EQU  $FC58
FDBE:         15 CROUT    EQU  $FDBE
FDED:         16 COUT     EQU  $FDED
0000:         17 *
0000:         18          DEND
0000:         19 *
----- NEXT OBJECT FILE NAME IS TEST STEP BY STEP V2.0.OBJ0
0900:         20          ORG  $0900
0900:         21 *
0900:20 58 FC 22          JSR  HOME
0903:A9 C1    23          LDA  #'A
0905:85 0C    24          STA  CA
0907:A9 C2    25          LDA  #'B
0909:85 0D    26          STA  CB
090B:A9 C3    27          LDA  #'C
090D:85 0E    28          STA  CC
090F:A9 C4    29          LDA  #'D
0911:85 0F    30          STA  CD
0913:A9 C5    31          LDA  #'E
0915:85 10    32          STA  CE
0917:20 2B 09 33          JSR  TEST1
091A:20 34 09 34          JSR  TEST2
091D:20 39 09 35          JSR  TEST3
0920:20 46 09 36          JSR  TEST4
0923:A5 10    37          LDA  CE
0925:20 ED FD 38          JSR  COUT
0928:4C D0 03 39          JMP  $03D0
092B:A5 0C    40 TEST1    LDA  CA
092D:20 ED FD 41          JSR  COUT
0930:20 BE FD 42          JSR  CROUT
0933:60       43          RTS
0934:A5 0D    44 TEST2    LDA  CB
0936:4C ED FD 45          JMP  COUT
0939:A9 ED    46 TEST3    LDA  #>COUT
093B:85 11    47          STA  $11
093D:A9 FD    48          LDA  #<COUT
093F:85 12    49          STA  $12
0941:A5 0E    50          LDA  CC
0943:6C 11 00 51          JMP($11)
0946:A9 FD    52 TEST4    LDA  #$FD
0948:48       53          PHA
0949:A9 EC    54          LDA  #$EC
094B:48       55          PHA
094C:A5 0F    56          LDA  CD
094E:60       57          RTS
```

On trial with honorware

Cliff McKnight argues that honesty is the best policy

WE'VE seen a lot of ideas cross the Atlantic in the last few years, and not all of them have been good ones – the originator of the fast food burger bar concept, for example, should be boiled in polysaturated fat in my opinion.

As Apple users though, we're used to getting a lot of our software from the States. Recently, a new kind has been making the crossing and it's based on a new idea in software distribution – honorware. The idea is simple: The software can be freely copied and distributed, but if you like it and use it you're asked to send a fee to the author.

It's a nice idea in many ways. For example, it means that you can try the software before parting with any money. If you don't like it, or it doesn't do what you want it to do, you can simply erase it from your disc and forget all about it. Also, the author gets real feedback about the quality of the program and how well it satisfies demand.

Open to abuse

Of course, all the advantages of the system depend on one thing – the user paying the fee. The system is easy to abuse when the software can be copied, and the name of honorware is an accurate one (even if the spelling is American).

It's a system that seems to have taken off in a big way in the Mac field, so the chances are that you've encountered at least one piece of honorware. It is also gaining ground in the PC world, although when I tried to get a copy of an honorware program from Guy Kewney through PCW, what I received was a letter from a company saying that the British weren't ready for this concept and offering to sell me the package.

Honorware shouldn't be confused with public domain software which anyone can copy and use. With honorware, the author retains all rights to the product. With many public domain programs, it's easy to see why they are so – nobody would pay for them anyway. With honorware this is certainly not the case.

Scott Watson's Red Ryder comms program is a case in point. I turned to it when Vicom couldn't cope with my Mac Plus, tried it out for a month and then paid my $45 to become a registered user. I've since had the Vicom upgrade but have not really found a reason to stop using Red Ryder and go back to Vicom. Red Ryder does everything I want and more.

Incidentally, probably the most powerful PC comms package is also a piece of "user supported software", another name for honorware. Procomm, like Red Ryder, has all the bells and whistles you could want, although being American neither will cope with the viewdata format of systems like Prestel.

We're hoping to give honorware packages the attention they deserve in future issues of *Apple User*, so what I'd like to do now is to mention some of the more interesting requests that I've seen in the "About . . ." pages of the programs.

Having just mentioned Prestel, I might as well begin with Paul Russell's Pretzel desk accessory which you can use to access viewdata systems. This is unusual in the honorware field because Paul is British.

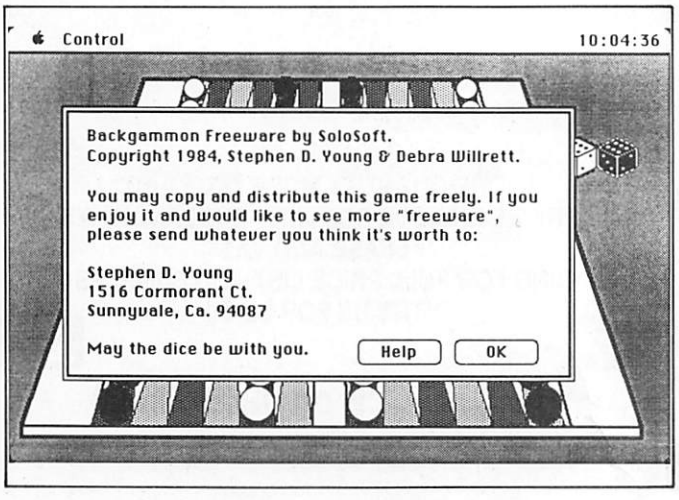
Online help

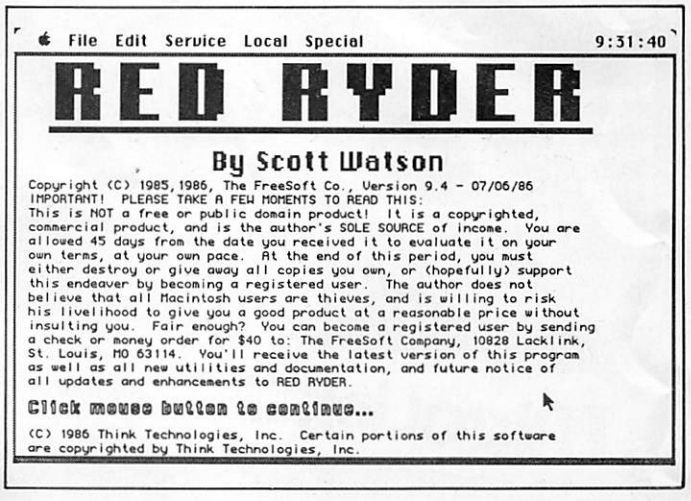
He doesn't actually ask you to send him any money either. He suggests that if you use the program you should send a donation to Oxfam or some other relief agency, a fine gesture that is to be applauded. What's more, if you send Paul the receipt for your donation you'll be notified of future updates and be eligible for online help.

In a similar vein, the We Are The Word desk accessory – a sort of Hangman game – asks you to contribute to USA for Africa or buy the We Are The World album.

The honorware version of Hangman written by Ken Winograd asks for "five or ten dollars (or even just a disc)". If enough people simply sent blank discs it would no doubt save him some expense over the next few months.

As you might expect, most of the programs come from America. The only British one I've seen apart from Pretzel is Patrick Buckland's Crystal Raider game. The startup ▷





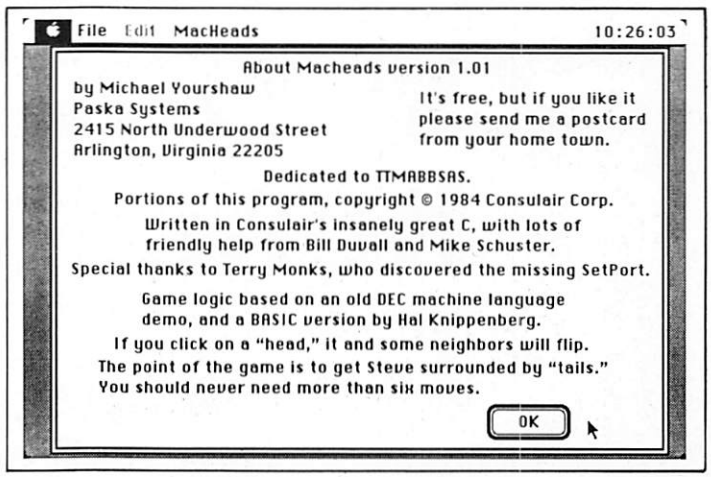
◁ screen promises to send the sequel, Crystal Quest, to users who send the £10 – I suspect that if I tried to tell you what Patrick suggests to those who don't pay up, it would fall foul of the editorial blue pencil.

I've also seen only one program of Australian origin, Thomas Schulze's MacPac DA implementation of PacMan. It's written for a US keyboard system so the keys are not as listed on UK machines, but for $5 it has to be the cheapest PacMan I've ever seen.

Feed the dog

The sole Canadian offering I've seen is the New Scrapbook DA. The author, Tim Wasko, asks you to send a "small contribution" if you like it, or a postcard with suggestions for what you'd like to see in future versions if you don't. Since it looks very nice, I might send him a card asking for future versions to work on machines not having 128k roms.

Several programs, like Stephen Young and Debra Willrett's Backgammon for example, suggest that you send whatever you think it merits. The strangest one of these I've come across is Stefan Bilaniuk's Double Apple DA which asks for a contribution of what you think it's worth to the Graduate Breakfast Fund in the Department of Math and Computer Science at Dartmouth College.

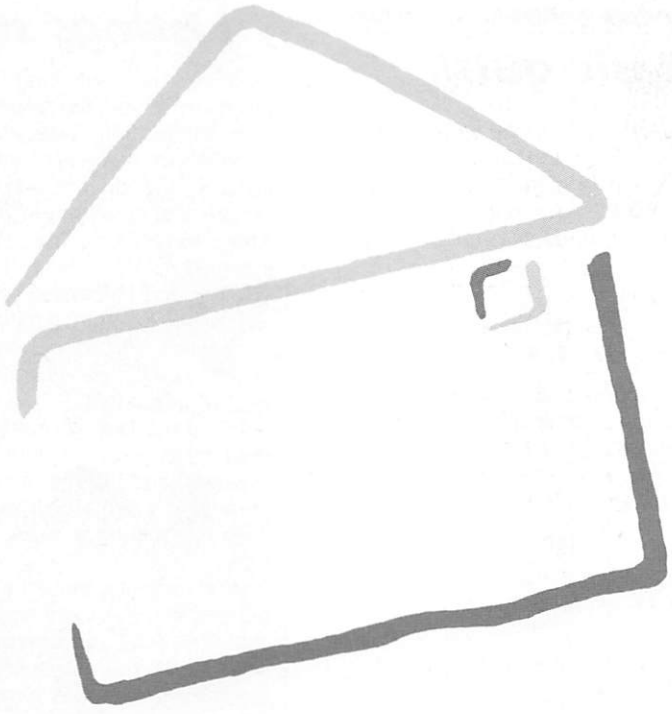


The prize for sheer persistence has to go to Al Evans' Cap'n Magneto game. This keeps interrupting the game with adverts suggesting that you "Make Mom happy" or ensure that the dog in the picture gets to eat by paying the honorware fee. What's more, with Macintalk in your system file, the program will speak the adverts to you too.

Finally, although it's not strictly honorware, my favourite is the request that accompanies Michael Yourshaw's MacHeads games. As he says, "It's free, but if you like it please send me a postcard from your home town".

I hope he has lots from Britain – after all, we are honest enough to make the honorware payments, right?

Next month I'll be reporting on what the honorware authors think of the system and the response they've been getting from us Brits. □



Apple III options

HAVING bought a business, I find myself the owner of an Apple III with dedicated green screen monitor and Drive 2, no printer and sundry discs – System Demonstration, System Utilities, System Utilities Data, Apple II emulation, Integer and Applesoft Basic, Apple Writer III and some others with handwritten labels.

Having had minimal experience, self-taught from books, on a Spectrum a few years ago, I tried some of these discs while waiting for my first copy of Apple User.

Volume 8 No 3 finally arrived, which I began to read with a view to finding software/hardware to use in my business. And – not a mention of an Apple III! Apple II, II+, IIe, II Europlus, IIgs, Macintosh – yes.

I did find the phone number for Apple Computer UK, which I rang, only to be told that no one does anything for an Apple III any more, but that the Emulation disc will allow me to use some II+ programs – very helpful, I don't think.

Any suggestions would be welcome. What options are open to me – can I modify or expand what I've got? Have tailormade programs written? But at what cost? Is there software/hardware available for the Apple III which is suitably versatile?

Or do I own a rare collector's item worth as much as the newest system which I can trade in for? – **M. Moyse, Suffolk.**

● From my small knowledge of it, the Apple III is a fine computer – it's just sad that it never really caught on. It seemed to be sandwiched between the much cheaper Apple II and the more expensive, but more capable Lisa. This in turn has been transformed into the now much more familiar Macintosh.

There was some software written specifically for the Apple III (and you seem to have some of this) but most users appear to have run the machine in Apple II emulation mode most of the time.

You don't say what you actually want to do with the machine. You have a word processor and a simple database program (albeit no printer) and really, unless you want more power, it's probably not worth your while spending any more money on new software.

Why don't you use what you have for a while and see how it goes? If you find you need a specific piece of software you can then go and find it, armed with the knowledge gained from some months experience.

Similarly, you will be better placed to see if you really want a new machine – and if so, which. – **Max Parrott**

Array check

*I HAVE just typed in the Calendar for 1987 (*Apple User, *December 1986) on my IIc and it says SYNTAX ERROR IN 2500. I have typed this line and checked the rest of the listing for errors, but still it says this. Do you know why?*

Earlier this year I bought Movie Maker but I can't back it up: If it gets ruined, is there anything I can do?

And I have a problem in Wizard & Princess: I have had to drop everything to cross a bridge and have got to a giant, and now I'm stuck. Please can you tell me how to get past – or am I attempting the impossible?

Finally, when I bought my IIc three years ago I was expecting 16 colours, but I can only use seven. Why is this? – **J. Shippam, Pagham, Sussex.**

● Your syntax error in line 2500 happens because there are numbers missing in the data lines above. Applesoft then expects to find a number but finds a string instead – and flags an error.

If you cannot spot the error by inspection, this kind of bug is best tracked down by temporarily inserting lines after each read line to write the data out again. This way, each array element may be checked.

To back up Movie Maker you need a bit copier – a piece of software which will make a copy of a protected disc – hopefully. Your only other options are to demand another copy from the supplier, or to try to de-protect the software before it crashes – both are hard.

Can anyone help with the Wizard & the Princess query?

Your IIc does have 16 colours – in the lo-res mode: In the hi-res mode there are seven colours.

Locked up

WHEN I started to read the letters section in the December issue of Apple User, *relating to the combination for the lock in Rungistan, I thought I had struck gold and that all my problems were over.*

This proved not to be the case, hence I'm asking for a little more help.

I have long since found all four sets of numbers – L14 21 4R L7 – but I am still having no success in opening the cabinet.

I first started to escape some 18 months ago, and during this time have filled many pages with different configurations of numbers and letters.

I had given up, and decided to remain a prisoner, but with the possibility of parole that the letter offered, I have a new lease of life.

The closest I have come to what I think is the combination is L74 R21 L14, so it could be the way I am inputting the information. Any small clue for an ageing inmate would be greatly appreciated! –**M.K. Blomley, Gwent.**

● The trick with the combination is to realise, as you have done, that the sequence will be L-R-L. However, the catch is that the bits of paper make you think there is an extra number.

The 4R is actually the other half of the L14, with the 4 being the same in each case. This gives L14R. Since you have a loose 21 and an L7, this means the combination becomes L14 R21 L7.

This should get you moving again. We trust you did a lot of reading while you were in prison... **Cliff and Denise McKnight**

Good graphics

THIS is to mention that I have used the graphics programs featured recently in Apple User *(the hi-res editor and Print Shop converter) and they work very well.*

I have also found useful the Dark Star System, which can freeze any graphic from ▷

◁ *any game, and with the help of the Monitor option save it on another disc under hi-res graphics with any name you wish (page 1 033 or 034 graphic).*

If you want to use part of the graphic with the Print Shop, you run Print Shop companion, then save the portion of graphic you like as page 1 graphic 004 under a different file name.

However, I use Newsroom more often than Print Shop, and I wonder if there is a way to move clip art graphics from games and incorporate them into Newsroom in the same way as I have been doing with Print Shop: This would be an addition to the clip art library for Newsroom. – **G. Inglott, Muscat, Sutanate of Oman.**

● I'm afraid we don't know how to move graphics into Newsroom, but we're sure that readers do. Perhaps someone will write in with details or a program?

Planet Suite

IS line 780 correct as printed in the program Planet Suite, Apple User, *January 1987? I've bunged a minus sign in before the INT and that seems to do the trick.*

I presume that this program should run on a IIc under Prodos, which it does anyway, but I do wish that the fact could be stated with the listing. – **MAG20272, MicroLink.**

● Sorry about that line – as you say, it should read:

```
780 B(I) = A(I) + (NT(I)
    - INT (NT(I))) * 430
```

Generally speaking, all the Basic programs we print run under Dos and Prodos and all Apple II machines unless we say otherwise.

Basic quirk

I HAVE recently spent many hours tracking down a bug that wasn't in my programming, but rather a quirk in the way that Apple Basic handles decimal-only numbers. The problem can be illustrated by the program:

```
10 X = .7
20 IF 10 * X <> INT (10 * X) THEN PR
   INT "THIS CAN'T BE RIGHT"
```

It would appear that Apple Basic is not always capable of rounding off errors created by conversion to binary. In the case of Line 10 it recognises the error, but when this result is then operated on by the

```
INT
```

it is not rounded up, and the error occurs.

The same kind of error occurs in the next line:

```
30 FOR J = 1 TO 10 * (8.7 - 8): PRIN
   T J : NEXT J
```

In fairness to Apple, I have to say that I have produced exactly the same errors on an Apricot, a BBC Micro and a Compaq. – **C. Hart, Wootton.**

● The quirk is a general one for all computers and computer languages, and as you say, arises because of the conversion of binary to decimal.

For safety, a programmer should never compare two floating point numbers directly. Always subtract one from the other and compare the ABS value of that against a small constant. If it is less than the constant, then consider the numbers equal.

Similarly, if you want to compare a floating point number with its INT, you should always round up the INT, using a decimal 5 with the appropriate number of zeroes in front of it and after the decimal point to set the accuracy you want – within the limitations of the language being used. – **Max Parrott.**

Listings style

STARTING this issue, long Basic listings will be presented in a new format. Each line will be split whenever a colon appears, so that each sub-statement appears on a new line, and the width of printing will be limited to 28 characters. Also, loops will be successively indented.

In addition, embedded control characters will be printed as their upper case equivalents. This should not cause any trouble as their use is generally obvious by their placement.

The most likely occurrences are in REM statements (Control+J to force linefeeds) and PRINT statements where Control+g (ring a bell) or Control+D (Dos control) have been buried in a string.

When entering these listings, remember that Return should be pressed only at the end of a Basic statement; that may be at the end of one or several lines of text. To put it another way, one line of Basic ends immediately before another one begins, and a line of Basic always begins with a line number.

Pascal, Forth and Assembly language listings will continue to be printed over the full width of the listing, which is generally 80 characters.

Church micros

I HAVE only just rediscovered a page I tore out of a November 1985 Apple User *in which an article, headed "Church moves into micros" mentioned the formation of the Christian Micro Users Association.*

Would you please be kind enough to locate the address of this group in your files and forward it to me. – **P. Cruice, Australia.**

● We're sorry, but we no longer have the address either, but if any reader cares to forward it we will pass it on to you. It may be useful to know that a bulletin board called Computers for Christ operates in the UK on 0395-272611 at 300 baud 8/1 bits.

CODING SOLUTION

Decoding key: 45 24 13 27

● Remove spaces

● Remove four characters from odd numbered lines, five from even numbered lines.

31S7131X9Z04CUC
0ZYHN0TK92SA1147
564Y2SZ3QLA4BT8
Y0JO1T449XBY40JQ

● Higher base is 27: Remove characters from R to Z.

317131904CC
0HN0K92A1147
56423QLA4B8
0JO1449B40JQ

● Odd numbered lines: Alternate base 13 (two character group), base 27 (four character group). Even numbered lines: Alternate base 27 (two character group), base 13 (four character group).

Line 1	317	Base 13 decodes to two character group EN
	13190	Base 27 decodes to four character group CODI
	4CC	Base 13 decodes to two character group NG
Line 2	0HN	Base 27 decodes to two character group DE
	0K92A1	Base 13 decodes to four character group CODI
	147	Base 27 decodes to two character group NG
Line 3	564	Base 13 decodes to two character group PR
	23QLA	Base 27 decodes to to four character group OGRA
	4B8	Base 13 decodes to four character group MZ
Line 4	0JO	Base 27 decodes to two character group EX
	0149B4	Base 13 decodes to four character group AMPL
	0JQ	Base 27 decodes to two character group EZ

Z corresponds to space, giving the decoded message:

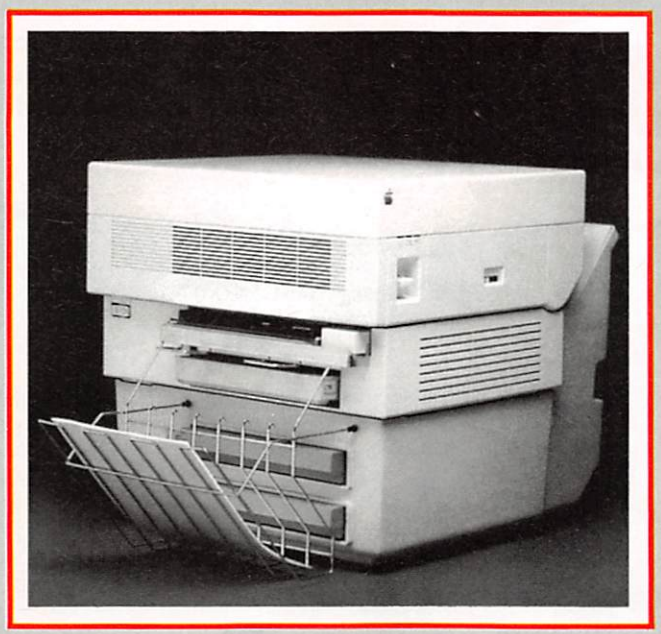
ENCODING DECODING PROGRAM EXAMPLE

Apple LaserWriter™ Users:

SEE US ON STAND 61F AT THE 1987 PC USER SHOW 30 June - 2 July Grand Hall, Olympia, London.

The more revisions and paper changes you make...

the more you need the extra capacity and flexibility of the Ziyad PaperJet™ 400/XP.



Call LaserGen NOW!
(04862) 28071

LASERWRITER IS A TRADEMARK OF APPLE

Available from your local Apple dealer or return the coupon below to LaserGen.

LaserGen

LASERGEN UK LTD
WBM HOUSE · HIPLEY STREET
OLD WOKING · SURREY GU22 9LQ
TEL: WOKING (04862) 28071
TLX: 858146

I would like to know more about the 400/XP for the Apple LaserWriter™

Name: ..... Position: .....

Company: .....

Address: .....

..... Telephone: .....

I have ..... Apple LaserWriters installed.

I also use ..... Canon/HP Laser Printers.

This could be your last chance to build up a library of knowledge from our back issues!

Bundle 1 January 1985 June 1985

Programming; Pascal large arrays and interrupts, Quicksort and binary search in Forth and Basic, Apple to Apple file transfer, add graphics commands to Basic, video techniques, RWTS & disc verify routine, reaction timer, Pascal directory/ date access, records in Pascal, graphics from light, split screen techniques, circle drawing, Basic music routines – Reviews; Clozemaster, Flex, Ormbeta Accounting, Sales Edge, Management Edge, Merl Modem, Intec drive, Vision 128/256, The Editor, Speedemon card, PFS on Mac, Workbench, Macputer IIc, Copytext, Mac Omnis 2, Tick-Tack translation, Mac Musicworks, – Spreadsheets; Targeting, estimating, Visicalc summarising, plus 11 to 13 of DIY Graphics and many games reviews.

Need a binder for your magazines? We'll send one for £4.95

**Inc. UK only postage & packing*

Only £4.50 per bundle

Bundle 2 July 1985 to December 1985

Programming; Pascal records in files and random access, binary file loader, ext 80-col. card memory usage, DOS 3.3 patches, scrolling fast on Wordstar, Basic restore utility, Pascal units, Basic line comparer, graphics dumps in imagewriter, Bomb-proof input, &DOS (dos routines directly) storing mc routines on language card, fast factorial function, lower case Pascal, assembly language interface to Pascal, free sectors under DOS 3.3 utility, Pascal speed up techniques, Block puzzle game in Forth – Reviews; Guide, Fingerprint, Printerupt, Mac Microplanner, Lotus Jazz, Layout packages, Ramworks 80-col. card, Computereyes & Magic digitizers, MEMDOS, Excel, Cirtech CP/M for IIc, DDTe debug card, Mactype, Mac the Knife fonts – Spreadsheets; general secrets, house/home, rosters – plus 14 to 17 of DIY Graphics, games reviews, Appletips.

Bundle 3 January 1986 July 1986

Programming; Pascal speed-up techniques, Heapsort in Forth and Basic, Pascal dynamic memory usage, Using Text Page 2, CP/M PIP patch, Machine code tracer utility, Applesoft lower case input routine, 3-D wire frame graphics, Scrolling hi-res pages, DOS patches, ProDOS disc routines, Hi-res picture editor Part 1, Easy renumbering. Reviews; Lawtant disc controller, Lemi Midi interface, Crunch, Mac+II, Cirtech and Tymac printer cards, Microsoft File, Ensemble, SpeedLoader, P-Tral, Ultraplan, Apple UniDisk, PlusWorks, BBC Basic, Snapshot Shuttle, Cirtech Flipper, Jeeves, GraphWorks, Resolution 64 card. Spreadsheets; Sales forecasting, Miles per gallon calculator, Annual salary budgets, Balance sheets in visual form, Checking electricity bills. Plus hardware projects, many games reviews and Appletips.

Bundle 4, July 1986 December 1986

Programming; Using CP/M's hidden areas, DOS lower case update, Retrieving Pascal disc directions, Hi-res picture editor Parts 2 & 3, Apple IIc graphics dump, Pascal ImageWriter dump, Automating WordStar, Simplifying graph production, Date stamping Apple IIc files, Mousekeeping with Pascal, ProDOS error messages, Date stamping DOS 3.3 files, ProDOS system file finder, Hi-res picture shrinker, Appointment program, Extra Basic commands. Reviews; MultiRam, FullText, SpectraGram colour card, Comprehensive Interface System, Aqsoft, Pinpoint, Cirtech Z80 board and CP/M Plus, MacroWorks, LogiMac, Peanut IIc external drive, Switchback, MacServe. Spreadsheets; Stock Market securities Parts 1 & 2, Cash flow budgets. Plus a look at the new Apple IIgs, Installing enhanced and standard roms in the IIe, many games reviews and Appletips.

BACK ISSUES

Catch up on articles you may have missed. Back issues from January 1987 are available at £1.75.

January 1987

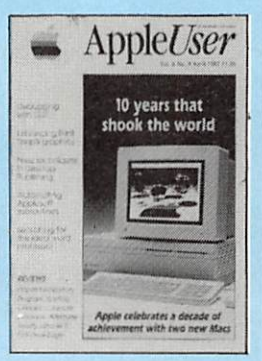
Review: Format-80 Scientific, Ramfactor and Multiram memory cards, Autoworks – Programs: Electronic Orrery, Text encoding – Tutorials: CP/M I/O devices & Pascal screen control – Interview with Steve Wozniak – Desk top publishing: Five pages covering the news on this up & coming market – Fun & Games: Toy Shop, Artic Fox, Decision in the Desert and Graphics Expander Vol. 1 – Utilities: Form making with MacPaint – Report on Apple UK trip to Kilimanjaro – Index to 1986 Apple User.

February 1987

Reviews; Transwarp, Multiscribe, AppleWriter (an overview), Dots Pfect and two hard discs – MacBottom and Hyperdrive – Programming: More printer control in Pascal, PIP & STAT in CP/M, & programs to give instant DOS 3.3 and auto line numbering under DOS & ProDOS – Interview: Second part with Steve Wozniak – Desk Top Publishing: Newspapers and Journals & The Wordsmith – Fun & Games: Standing Stones, Mind Pursuit, Uninvited, Mask Parade and The Pawn all reviewed.

March 1987

Reviews: Micol basic, ComicWorks and GraphicWorks for budding cartoonists – Programming: Device assignments in CP/M, file editing in Pascal, coding and decoding and playing Patience – Utilities: Booting Pascal 1.3 and customising CIA Files – Desktop Publishing: Graphics Factory visited and an update on the latest hardware and software – Fun and Games: Shanghai, 221B Baker Street and Crosscheck reviewed – PLUS all the latest Apple News and your letters.



April 1987

Reviews; Pinpoint Pop up Spelling Checker, Pinpoint Ram Enhancement, Gutenberg Word Processor – Programming; Using DDT in CP/M, Pascal Printer Control Unit in action – Utilities; Making more of Print Shop's Graphics, Encoding continued – Application; Low-cost Image Analysis – Desk Top Publishing; Macs in the newspaper industry, an introduction to DTP, book on PageMaker techniques – Fun & Games; F-15 Strike Eagle, Crusade in Europe, Alternate Reality, The City, Hacker II – Feedback.

May 1987

Reviews; Mousestuff for Pascal, Swyftware on disc, Multiscribe 2 – Fun & Games; Prince (colour printer kit), Brian Clough's Football Fortunes, Hollywood Hijinx, Starglider and King of Chicago – Programming; Pascal File Control System, CP/M non-disc calls, Multi-choice quiz game – Utilities; Toolkit Assembler from the Flipper, Graphics library cursor routine, encoding techniques – Desk Top Publishing; Quick Print Franchising, Fancy Fonts – PLUS all the latest Apple news and your letters.

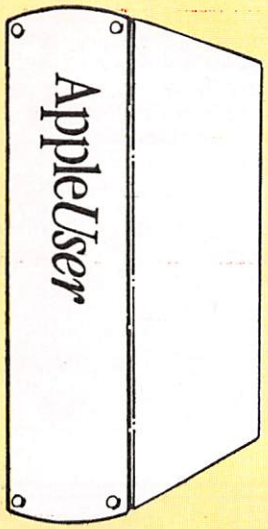
Apple User binder

Your Apple User is the ideal source of reference for every user of Apple computers. Keep your magazines tidy and in tip-top condition by using our top quality binder.

The *Apple User* binder holds 12 issues. Each binder is embossed with the *Apple User* Logo.

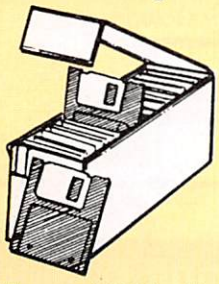
Only £4.95

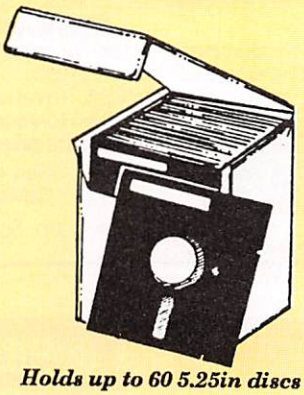
See order form for prices outside the U.K.



FREE Secure storage for your discs (WORTH £4.95)

...when you take out a subscription

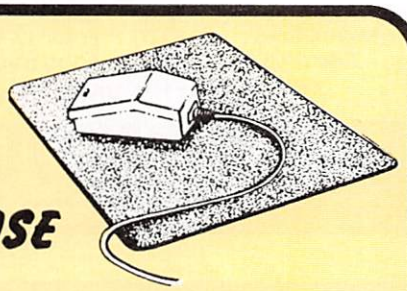




Holds up to 50 3.5in discs ***Holds up to 60 5.25in discs***

This is the ideal storage medium for your discs – a luxury padded pvc container, strongly made to protect your delicate programs from harm.

FREE MAT FOR YOUR MOUSE



(WORTH £4.95)

...when you take out a subscription to Apple User

Our top-quality Mouse Mat with its specially-Designed perfect-grip surface, provides the ideal desktop enviroment for your Mouse.

- ✧ **Ensures much smoother movement**
- ✧ **Gives super positive control!**
- ✧ **Protects highly polished tabletops**

Your Mouse Mat comes FREE if you subscribe using this form. Non-subscribers can obtain one by sending £4.95

AppleUser

ORDER FORM

Valid to June 30, 1987

Offers subject to availability
All prices include postage, packing and VAT

All overseas items despatched by air mail

£

New subscriptions

UK £18	*1001*	
Europe & Eire £23	*1004*	
Overseas surface £25	*1000*	
Overseas Air mail £38	*1005*	

Commence with _____ issue

Subscription renewals

UK £18	*1002*	
Europe & Eire £23	*1006*	
Overseas surface £25	*1003*	
Overseas Air mail £38	*1007*	

* Free with subscription – tick ONE box only:
Mousemat *1034* ☐ Disc Box 5.25" *1036* ☐ Disc Box 3.5" *1038* ☐

Disc Storage Box NEW

5.25"	£4.95	*1037*	
3.5"	£4.95	*1039*	

Mouse Mat NEW

£4.95 *1035* ☐

Silicon Dreams or Jewels of Darkness

	With Subs*	Without Subs				
Apple IIe						
Silicon Dreams	£12.95	£14.95	*1014*		*1015*	
Jewels of Darkness	£12.95	£14.95	*1016*		*1017*	
Both	£23.90	£27.90	*1018*		*1019*	
Macintosh						
Silicon Dreams	£12.95	£14.95	*1020*		*1021*	
Jewels of Darkness	£12.95	£14.95	*1022*		*1023*	
Both	£23.90	£27.90	*1024*		*1025*	

Add £3 for Europe (inc Eire)/£5 for overseas
(If ordering both add £6 for Europe (inc Eire)/£10 for Overseas

Starglider

	With Subs*	Without Subs				
Apple IIe	£9.95	£14.95	*1008*		*1009*	

Elite

	With Subs*	Without Subs				
Apple IIe/IIc	£9.95	£14.95	*1087*		*1089*	

Shanghai

	With Subs*	Without Subs				
Apple IIe	£14.99	£19.99	*1010*		*1011*	
Macintosh	£14.99	£19.99	*1012*		*1013*	

Bargain Bundles of Apple User back issues NEW

£4.50 UK, £7.50 Europe (inc Eire)
£10.50 Overseas Air mail

Bundle 1 Jan – June '85	*1030*		
Bundle 2 July – Dec '85	*1031*		
Bundle 3 Jan – June '86	*1032*		
Bundle 4 July – Dec '86	*1033*		

£1.75 UK, £2.25 Europe,
£2.75 Overseas Air mail

JAN 1987	*1209*	
FEB 1987	*1210*	
MAR 1987	*1211*	
APR 1987	*1212*	
MAY 1987	*1213*	

Binder

£4.95 UK £7.95 Europe (Inc Eire) £11.95 Overseas *1067* ☐

**Only if accompanied by a subscription order or renewal*

Add £2 for Europe (incl. Eire), £5 for Overseas to all prices unless otherwise indicated

Payment: please indicate method (✓) TOTAL _____

☐ Access/Mastercard/Eurocard/Barclaycard/Visa

No. | | | | | | | | | | | | | | | | |

☐ Cheque/Eurocheque made payable to Database Publications Ltd.

Exp. date _____

Name _____ Signed _____

Address _____

_____

_____ Tel: _____

Send to: *Apple User*, FREEPOST, Europa House, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.
(No stamp needed if posted in UK) *Please allow 28 days for delivery*

Order at any time of the day or night

Telephone Orders:	*Orders by Prestel:*	*MicroLink/Telecom Gold*
061-429 7931	Key *89, then 614568383	72:MAG001

Don't forget to give your name, address and credit card number

ENQUIRIES ONLY: 061-480 0171 9am-5pm

AU6

ALPHATRONICS

(Please note all items are compatible with all versions of Apple DOS, ProDOS, CP/M and Pascal unless otherwise stated)

RAM MEMORY EXPANSION

16K RAM card II/II Europlus/II+, gives 48K CPUs the extra memory to run CP/M and Pascal op systems £25

64K/80 COLUMN extended text card IIe, gives 55K with Appleworks, 60K ramdisk with ProDOS £25

128K RAM card II/II Europlus/II+/IIe £69
1MB RAM expansion/80 COLUMN for IIe RAMWORKS compatible, RAM disk facility – the latest addition to our range. £89
(1MB RAM £89 EXTRA).

MASS MEMORY

360K half height floppy drive, pancake motor high quality direct drive XM4 mechanism – Japanese origin £89

13/16 sector, twin port disc drive controller £25
(suitable for use with both standard Apple or half height drives)

Accelerator for IIe, process 3.5 times quicker £150

Z-80 coprocessor card, to run CP/M software £25

PRINTER/SERIAL INTERFACES

Epson/Centronics incl cable, supports all standard control codes and graphic manipulation features £29

Grappler + compatible incl cable £29
Grappler + compatible with 64K printer buffer incl cable £59

RS232 serial printer card £29
Serial plus card – suitable for use with a modem, with software selection of protocol and full Apple super serial emulation £49

80 COLUMN

80 column Videx compatible for II+/II/Europlus
(has built in 40 col/80 col softswitch) £35

80 column extended text card for IIe, with 64K memory, gives 55K with Appleworks, 60K ramdisk with ProDOS £25

80 column text card for IIe £19

Keypads/keyboards

IIe keypad, 16 keys: plugs directly into port on IIe mother board, has following keys
+ - * =. ENTER 0 1 2 3 4 5 6 7 8 9 £19
Replacement keyboard for IIe £49

Other hardware for II+ and IIe
Heavy duty power supply 7.5A output £49

ALPHATRONICS
Wolverley House, 18 Digbeth,
Birmingham B5 6BJ
Phone: 021-643 9866 (4 lines)
All prices exclude VAT
Delivery: £3.00 + VAT for orders less than £100, over £100 delivery free
(Items by carrier £8.95 + VAT extra)

ADVERTISERS' INDEX

EX DEMO – SECOND HAND

USED APPLE IIe's £175
HARD DISC Symbiotic - Ice from £150
USED APPLE III (Ink Profile) £795

Amstrads/Tandon	£POA
Ex demo Mac +	£1450
Mac 512/400	£900

SOFTWARE

Open item purchase – sales – nominal – invoicing – stock or job costing and cash book.
Special software for Airfreight, Travel agency, Garages etc.
All systems can be customised to your requirements.
Ring for prices

SOUTHERN COMMERCE COMPUTERS LIMITED

New - Used systems
Software based on Omnis III

219 Croydon Road, Caterham, Surrey. Tel: 0883 48919
Authorised Apple Dealer

Disks & Ribbons

DISKS

3.5" DS unbranded	£1.26
5.25" Memorex branded SS/DD	£0.68
5.25" 96 tpi DS, unbranded, labels etc	£0.43
SEE 10 Storage box 5.25" or 3.5"	£1.85

RIBBONS

Imagewriter	£2.65	Cannon PW 1080A	£2.69
Citizen 120D	£3.50	Citizen MSP 10	£4.90
Epson MX/RX/FX80	£2.60	Epson LX 80/86	£2.50
Epson MX/RX/FX100	£3.10	Kaga Taxan	£2.69

Please add 15% VAT, carriage free

T-Systems Ltd

The Signal Cabin, 61 High Street, Orpington, Kent BR6 0JF
Tel. 0689 22196

MACINTOSH SE – £1995
(Part Exchange welcome on your Mac Plus)

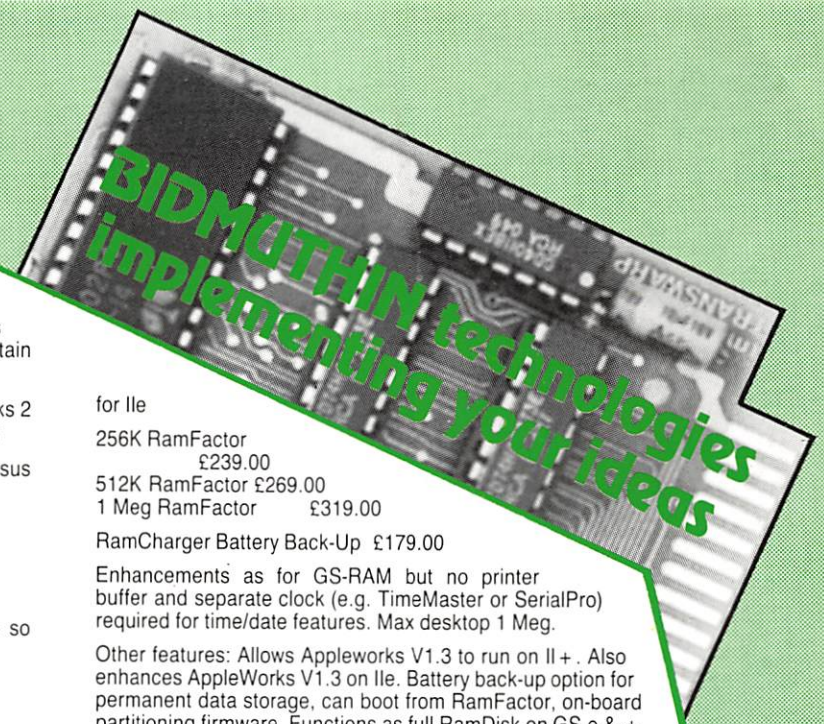
Macintosh SE/HD	£2500
Apple IIGS 512K Colour	£1075
Imagewriter 11	£345
Hard Disk 20 SCSI	£795
LaserWriter Plus	£4200

Please add VAT to all prices

Large stock of new & ex-demo Apple computers for sale

COMPUTER LINE

052524 243



APPLEWORKS AND IIGS APPLEWORKS 2 EXPANDER

The Appleworks 2 Expander is an enhancement program for Appleworks 2 which enhances and expands Appleworks 2 when used with certain memory cards.

Here's a list of the AppleWorks enhancements created by the AppleWorks 2 expander when used with GS-RAM and GS-RAM Plus cards on the IIGS:

* Word Processor and database Clipboard expanded to 2,042 lines (versus 250)
* Up to 22,600 lines in the Word Processor (versus 7,250 lines)
* Up to 22,600 records in the Data Base (versus 6,350 records)
* Variable size printer buffer
* On-screen time display
* Automatic time/date entry into the database
* Print-file cache (i.e. ALL of AppleWorks is loaded into memory — so doesn't access the disk to print)
* Expander options menu
* Multiple disk file-saving capability
* AppleWorks Desktop expansion to 8 Meg

Other features: GS-RAM and GS-RAM Plus are totally compatible with and surpass the Apple GS Memory card standard.

The Appleworks 2 expander is supplied FREE with GS-RAM and GS-RAM Plus.

Prices (excluding VAT)

GS-RAM 256K	£169.00
GS-RAM 512K	£209.00
GS-RAM 1 Meg	£269.00
GS-RAM 1.5 Meg	£319.00
GS-RAM Plus 1 Meg	£399.00
GS-RAM Plus 2 Meg	£629.00
GS-RAM Plus 3 Meg	£849.00
GS-RAM Plus 4 Meg	£1059.00
GS-RAM Plus 5 Meg	£1259.00
GS-RAM Plus 6 Meg	£1419.00
GS-RAM Plus7-8 Meg	£CALL
256K Upgrade Kit	£40.00
1 Meg Upgrade Kit	£240.00

The AppleWorks 2 Expander also works with, and is supplied FREE with these other memory cards:

for IIC

Z-RAM Ultra 2 — 256K	£269.00
Z-RAM Ultra 2 — 512K	£309.00
Z-RAM Ultra 2 — 1 Meg	£389.00
Z-RAM Ultra 3 — 256K	£329.00
Z-RAM Ultra 3 — 512K	£369.00
Z-RAM Ultra 3 — 1 Meg	£449.00
65C816 16-Bit Option	£79.00

Enhancements as for GS-RAM and GS-RAM Plus but maximum desktop is 727K (1 Meg Z-RAM Ultra) and printer buffer is fixed at 64K. The in-built clock also provides file date/time stamping as well as other time options. AppleWorks is fully pre-loaded into RAM. Also provides most of the enhancements for ALL earlier versions of AppleWorks.

Other facilities: 16-Bit option, clock.
Z-RAM Ultra 3 includes Z-80 co-processor and CP/M.

for IIe

256K Ramworks III	£199.00
512K Ramworks III	£239.00
1 Meg Ramworks III	£319.00
1.5 Meg Ramworks III	£469.00
3 Meg Ramworks III	£1,299.00

Enhancements as as for Z-RAM Ultra, but separate clock (e.g. TimeMaster or SerialPro) required for time/date features and printer buffer only works with Super Serial Card (or SerialPro).

Other features: 80 column display built in. (Replaces 80 col card). RGB option.

for IIe

256K RamFactor £239.00
512K RamFactor £269.00
1 Meg RamFactor £319.00

RamCharger Battery Back-Up £179.00

Enhancements as for GS-RAM but no printer buffer and separate clock (e.g. TimeMaster or SerialPro) required for time/date features. Max desktop 1 Meg.

Other features: Allows Appleworks V1.3 to run on II+. Also enhances AppleWorks V1.3 on IIe. Battery back-up option for permanent data storage, can boot from RamFactor, on-board partitioning firmware. Functions as full RamDisk on GS,e & +.

OTHER HARDWARE

Heavy Duty Power Supply (e/+)	£75.00
IIc System Clock (c)	£49.00
Phasor Music/Speech Synthesizer (e/+/GS)	£179.00
Pinpoint Apple IIe Enhancement Kit (e)	£29.00
Pro-App 20 Mb Hard Disk (IIc/GS/e, Mac, Mac+)	£995.00
Serial Pro — Serial Card + Clock (e/+/GS)	£139.00
TimeMaster II H.O. Clock (e/+)	£99.00
TransWarp Accelerator (IIe/+)	£279.00
Viewmaster 80 (80 cols on II+)	£139.00
Z-80+ card inc. CP/M (GS/e/+)	£139.00

IIGS SOFTWARE

AutoWorks 2.0 — Macros etc. (GS/e/c)	£49.00
DazzleDraw — dble hi-res graphics (e/c)	£69.00
Document Checker — fast spell checker (GS/e/c)	£69.00
FontWorks — for AppleWorks (GS/e/c)	£49.00
InfoMerge — for AppleWorks (e/c)	£29.00
Keyplayer Macros (reqs Pinpoint) (GS/e/c)	£49.00
MultiScribe — GS Wordprocessor (GS)	£99.00
MultiScribe 2.0 Wordprocessor (e/c)	£69.00
PinPoint 2.0 — Desktop accessories (GS/e/c)	£89.00
PinPoint Toolkit (inc RunRun) (GS/e/c)	£69.00
Point-to-Point — Communications (GS/e/c)	£99.00
ProBasic w. Program Writer (GS/e/c)	£49.00
ProFiler — Database (GS/e/c)	£99.00
Quark Catalyst 3.0 — Mac-like Desktop mgr (e/c)	£49.00
RamDrive (Dos3.3/ProDos/Pascal/CP/M) each (e/c)	£29.00
Ram Enhancement/Management Kit (e/c)	£29.00
RunRun — Desktop Manager (e/c)	£49.00
Spelling Checker (reqs Pinpoint) GS/e/c)	£69.00
TopDraw — Graphics for GS (links w. MultiScribe)	£99.00
VIP Professional — e/c (Lotus 123 on e/c)	£199.00
VIP Professional — GS (Lotus/Excel on GS)	£249.00
Visicalc Expander Disk (e/c)	£29.00
Visualiser — e/c — Graphs AppleWorks (e/c)	£79.00
Visualiser — GS — Graphs AppleWorks (GS)	£89.00

All products are also available from your local dealer.
All products carry a ten-day no-quibble "money back if not delighted" promise plus a one year guarantee.
Ordering Information Add £1.00 P&P per order. Add VAT at 15%

VISA

BIDMUTHIN technologies

P.O. Box 264, Harrow, Middlesex, HA3 9AY
Tel: 01-907 8516
Telex: 8950511 ONEONE G
(Ref: 22554001)

The Drives that move your Mac up a gear

FREE MacServe with the AST 4000

UNQUALIFIED PERFORMANCE – The AST-4000 70Mbyte hard disk with backup has a 30 millisecond average disk access time, 5Mbyte per minute tape transfer rate, and expandability to 370Mbyte.

AST-2000 and AST-4000 provide the ideal, secure answer to all your information storage requirements. Being backed by AST's experience and reputation in add-on PC products you can expect the best in the business. Just as you can also expect the best in service from P&P Micro's excellent, time-proven dealer supported network.

The Macintosh Computer – a great machine that is exciting the imagination of people all over the World. And now the Mac's performance has been improved by the introduction of AST disk drives with backup, AST-2000 and AST-4000.

The AST-2000 is a SCSI 20Mbyte hard disk and 20Mbyte cartridge tape drive. The tape appears as a disk icon allowing "disk to disk" style file transfer. It provides fast, reliable data storage, data management and data backup. Its capacity and response time allow the user to manage extensive data bases, run sophisticated applications programs and use the AST-2000 in a file and resource sharing environment of AppleTalk networking products.

Return the cut-out coupon for more information and details of your local dealer.

Please send me further details of AST-2000/4000 ☐
Apple Macintosh ☐ Other APPLE products ☐
Other AST products ☐ and the name of my local dealer.

Name .....

Company .....

Address .....

.....

.....

Tel .....

Return to P&P Micro Distributors plc , FREEPOST,
Carrs Industrial Estate, Haslingden, Rossendale, Lancs.
BB4 5BR. Tel: 0706 217744 Telex: 6455740 PETPAM G. AU/6/87

AST EUROPE Ltd

P&P MICRO DISTRIBUTORS PLC

Todd Hall Road, Carrs Industrial Estate, Haslingden, Rossendale, Lancs BB4 5HU. Tel: 0706 217744 Telex: 635740 PETPAM G
1, Gleneagle Road, London SW16 6AY. Tel: 01 677 7631 Telex: 919220 PPMICR G
Dale Street, Bilston, West Midlands, WV14 7JY. Tel: 0902 43913

P & P | MICRO | DISTRIBUTORS | PLC